THE APPLICATION OF DIGITAL COMPUTERS TO ROOT-LOCUS ANALYSIS

A THESIS

Presented to

the Faculty of the Graduate Division

By

Cecil Orie Alford

In Partial Fulfillment

of the Requirements for the Degree

Master of Science in Electrical Engineering

Georgia Institute of Technology

August 1959

5 D
12 RT

"In presenting the dissertation as a partial fulfillment of the requirements for an advanced degree from the Georgia Institute of Technology, I agree that the Library of the Institution shall make it available for inspection and circulation in accordance with its regulations governing materials of this type. I agree that permission to copy from, or to publish from, this dissertation may be granted by the professor under whose direction it was written, or, in his absence, by the dean of the Graduate Division when such copying or publication is solely for scholarly purposes and does not involve potential financial gain. It is understood that any copying from, or publication of, this dissertation which involves potential financial gain will not be allowed without written permission.

_____

_____

THE APPLICATION OF DIGITAL COMPUTERS

TO ROOT-LOCUS ANALYSIS

Approved:

[redacted]

Date of Approval: August 10-1959

ACKNOWLEDGEMENTS

The author wishes to express his sincere appreciation to Dr. Frank O. Nottingham for his assistance as thesis advisor and to the members of the reading committee, Drs. W. F. Atchison and W. J. McKune.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF ILLUSTRATIONS

LIST OF ILLUSTRATIONS

TERMINOLOGY

Control System

An assembly of devices capable of forcing one quantity to follow the variations of another quantity.

Signal

A variable quantity in a system, such as a voltage, a force, or an angle.

Input: θ_i, $\theta_i(s)$, $\theta_i(t)$

The arbitrary signal which is impressed on a system, the independent variable.

Output: θ_o, $\theta_o(s)$, $\theta_o(t)$

The signal produced by the system. Often it duplicates the input signal as closely as can be achieved.

Error: θ_e, $\theta_e(s)$, $\theta_e(t)$

Often the difference between the input and the output signal; in general, the error is the difference between the actual output and the desired output.

Frequency-Response

The characteristic of a system described by the transmission of sinusoidal signals of various frequencies.

Transient-Response

The natural response of a system due to any disturbance such as a sudden change in the input.

Linear System

A system whose characteristics are independent of the amplitude of the signal or of time.

Transfer Function: $\frac{\theta_o}{\theta_e}(s)$, $\frac{\theta_o}{\theta_i}(s)$, $\frac{V_2}{V_1}(s)$, $F(s)$, $G(s)$

The description of a component as the ratio of output to input of that component for the case of an exponential signal , e^{st}, being transmitted.

Root-Locus

For a system whose characteristic equation is $KG(s) + 1 = 0$ the root-locus is a plot in the "s" plane of the values of "s" which make G(s) a negative real number. A point along the locus is a root if the gain, K, is selected to make $K\,G(s) = 1$.

Zero: z_1, z_2

A value of "s" which makes a given function, usually F(s), zero. On root-locus plots, a zero is denoted by a small circle (o).

Pole: p_1, p_2, p_3

A value of "s" which makes a given function, usually F(s), infinite. On root-locus plots, a pole is denoted by a cross (x).

Phase Angle: $\angle F(s)$, $\angle \overline{A}$, $\angle \overline{B}$, ϕ_1 ϕ_2, ϕ_3, θ

The angular difference between the output and input of any device or the displacement of a vector from some reference.

Magnitude: $|F(s)|$, $|\overline{A}|$, $|\overline{B}|$

The ratio of the absolute values of the output to the input of any device or the absolute value of a vector when compared to some unit vector.

Complex Plane: "s"

A complex variable is represented by "s" where $s = \sigma + jw$. The entire domain of all "s" comprises the complex plane.

Real Variable: σ , σ_1, σ_2

A single line in the complex plane represented by the real axis.

Point of Breakway:

The point at which the root-locus departs from the real axis.

SUMMARY

The root-locus method is only one of several methods for analyzing servomechanisms and circuit theory problems. At least two mechanical devices, the Spirule and the Complex Plane Analyzer, have been developed as aids in solving for the root-locus. The problem presented was to obtain the transfer function of a given servomechanism using frequency-response techniques, to sketch the root-locus by established rules and methods, and then to compute the root-loci using digital computing techniques.

Frequency-response data for the uncompensated system was obtained using a closed-loop, low-gain feedback system. From this data the asympotic approximations were constructed and the open-loop transfer function obtained. Derivative compensation was then taken from the motor armature voltage and applied to the amplifier grids. The frequency-response data obtained led to the open-loop transfer function for the compensated system.

After a sketch of the root-locus was made from the transfer function, the International Business Machines 650 Digital Computer was used to solve for the root-loci, using the Bell Telephone Laboratories General Purpose System. Programs were written to solve for complex points along the root-locus for transfer functions containing three poles,

four poles-one zero, and three poles-one zero.

The essential problem was one of solving the equation for the point "s_j" in the "s" plane at which

$$\sum \angle \text{vectors from zeros to "}s_j\text{"} \cdots - \sum \angle \text{vectors from poles to "}s_j\text{"} = 180^\circ + n(360^\circ)$$

where "n" is equal to any integer or zero. This equation was solved by successive approximations, using a linear approximating method, until the error was reduced below some prescribed value. Then the value of gain at the point was determined by

$$K = \frac{\text{Product of the magnitude of the vectors}}{\text{Product of the magnitude of the vectors}} \cdots \cdots \frac{\text{from poles to "}s_j\text{"}}{\text{from zeros to "}s_j\text{"}} .$$

The intervals between successive points was chosen arbitrarily as 0.5. The number of points calculated varied with the particular problem. Tables are given showing the programs used, flow charts illustrating the logical method of solution and tables containing the computed information. Tables are also given indicating the rapidity of convergence at selected points along the root-locus.

Two programs were written to solve for the gain along the real axis, since the points contained in the root-locus were known here. These programs, flow charts and the computed information are given in the tables.

The method of solution is rapid and accurate; however, programming time is in general long and tedious. If only one problem were to be solved it would hardly be practical to use a digital computer. The method might be used to advantage in solving several problems of the same type, or in investigating changes in the critical frequencies for one particular problem.

CHAPTER I

INTRODUCTION

History. --Several methods have been developed for investigating the performance of servomechanisms. These include the classical or differential equation method, the frequency-response method, the root-locus method and other methods which have been developed more recently. Each of the methods has its advantages and disadvantages. From a control system designer's point of view, the root-locus method offers many advantages once the locus of roots is obtained. The single disadvantage is the difficulty of obtaining this locus of roots. A method of removing this difficulty is of primary importance in this paper.

The root-locus method was first presented by W. R. Evans in the article, "Graphical Analysis of Control Systems", (Reference 1). It is described by him as "a graphical method for finding the roots of the characteristic equation of the form $F(s) + 1 = 0$, in which $F(s)$ is in factored form". Single loop control systems have the specified form, where $F(s)$ is the direct transfer function around the loop for a system with negative feedback. The factors contained in $F(s)$ are the transfer functions of the components of the system. If the transfer function of any component is not known then an approximation can be used if frequency-response data for the component is available. After the

transfer function is obtained in analytic form, the zeros and poles of F(s) are plotted in the complex plane and used as the basis for sketching the locus of roots versus the loop gain as variable. A graphical picture is then presented of the variation in the closed loop poles as a function of the open loop poles and gain. When properly interpreted both the transient and frequency responses can be obtained from this locus of roots. The method is not limited to single loop control systems but can also be applied to more complex systems or to problems of a different type if the equation can be put in the specified form. (2)

One of the significant differences in the root-locus method lies in the fact that the transfer functions of the individual components remain separate, whereas in other methods each term contains a combination of parameters. This is important in control-system design, since the effect of changing certain parameters must be known. Therefore any method which determines the locus of roots should do so in a way which preserves the effects of these parameters. From these considerations it is evident that the usefulness of the root-locus method in the design of feedback control systems depends directly on the ease with which the loci can be constructed.

The Root-Locus Method.*--The basic elements of the root-locus method are illustrated by consideration of the system of Figure 1, page 8, a

---

*The material presented in this section is taken largely from Control System Synthesis by J. G. Truxal. New York: McGraw-Hill Book Company 1955, pp. 223-226

single loop feedback control system with the forward transfer function F(s). The closed-loop system function is given by the relation

$$\frac{\theta_o(s)}{\theta_i(s)} = \frac{F(s)}{1 + F(s)} \tag{1}$$

In the usual feedback control system, F(s) is a rational algebraic function, the ratio of two polynomials in "s".

$$F(s) = \frac{p(s)}{q(s)} \tag{2}$$

Substitution of equation (2) in (1) gives

$$\frac{\theta_o(s)}{\theta_i(s)} = \frac{p(s)/q(s)}{1 + p(s)/q(s)}, \tag{3}$$

$$\frac{\theta_o(s)}{\theta_i(s)} = \frac{p(s)}{p(s) + q(s)} \tag{4}$$

Thus for the system of Figure 1, page 8, the zeros of the closed-loop system function are identical with the zeros of the open-loop transfer function, and the closed-loop poles are the values of "s" at which $p(s)/q(s) = -1$. The basic difficulty with design in terms of the Laplace transform arises because the poles of the closed-loop system function are the zeros of the polynomial $p(s) + q(s)$. In any but the simplest cases, the evaluation of these poles for a given p(s) and q(s) is a tedious job; if, in addition, the motion of the poles with changes in a system

design parameter is desired, straightforward calculation becomes impractical.

The root-locus method is a graphical technique for determining the zeros of $p(s) + q(s)$ from the zeros of $p(s)$ and $q(s)$ individually. Then, if a system parameter is varied, the corresponding changes in the zeros of $p(s) + q(s)$ are investigated

The root loci consist of all points in the "s" plane at which the phase of the open-loop transfer function, $-p(s)/q(s)$, is 0° [or $0^\circ + n(360^\circ)$, where n is any interger or zero] . Or the root loci are the points at which the phase of $p(s)/q(s)$ is $180^\circ + n(360^\circ)$, where n is any interger or zero. An alternate definition is that the root loci are plots of the variations of the poles of the closed-loop system function with changes in the open-loop gain.

The root loci are thus the roots of the equation $F(s) = -1$, as a function of the open-loop gain. Since $F(s)$ is a complex function, one condition which must be satisfied is that the root loci constitute all points in the "s" plane at which

$$F(s) = 180^\circ + n(360^\circ) \tag{5}$$

where "n" is any interger, including zero. Equation (5) is the basis for all rules and techniques for construction of the root loci.

$F(s)$ at any specific point in the "s" plane is conveniently measured in terms of the angles contributed by the various poles and zeros.

Written in factored form, a typical F(s) for a feed back control system is

$$F(s) = \frac{K(s + z_1)(s + z_2)}{s(s + p_2)(s + p_3)(s + p_4)} \tag{6}$$

At the point s_1, F(s) takes the value

$$F(s_1) = \frac{K(s_1 + z_1)(s_1 + z_2)}{s_1(s_1 + p_2)(s_1 + p_3)(s_1 + p_4)} \tag{7}$$

The value of $F(s_1)$ can be expressed in terms of the vectors shown in Figure 2, page 9, as

$$F(s_1) = \frac{K \bar{A} \bar{B}}{\bar{C} \bar{D} \bar{E} \bar{F}} \tag{8}$$

where the zeros of the function are represented on the "s" plane by small circles and the poles of the function are represented by crosses. The angle of $F(s_1)$ is simply determined by the angles of the vectors:

$$\angle F(s_1) = \angle A + \angle B - \angle C - \angle D - \angle E - \angle F \tag{9}$$

Thus, construction of the root loci involves the determination of those points, "s_j", in the "s" plane at which

$$\sum_{K=0}^{n} \angle Z_{K \text{ to } S_j} - \sum_{L=0}^{m} \angle P_{L \text{ to } S_j} = 180^\circ + n(360^\circ) \tag{10}$$

where $\sum \angle Z_{K \text{ to } S_j}$ is the sum of the angles of the vectors from the zeros to the point "s_j" and $\sum \angle P_{L \text{ to } S_j}$ is the sum of the angles

of the vectors from the poles to the point "s_j". The second condition for "s_j" to be a point of the characteristic equation is that the magnitude of this function be unity. Referring to equation (8) the value of K which will make $F(s_1)$ equal to unity can be found from

$$\left|F(s_1)\right| = 1 = K \frac{|\bar{A}|\ |\bar{B}|}{|\bar{C}|\ |\bar{D}|\ |\bar{E}|\ |\bar{F}|} \tag{11}$$

Thus, the value of gain for the point "s_j" in the "s" plane at which equation (10) is satisfied is

$$K = \frac{\prod_{L=0}^{m} P_L \text{ to } s_j}{\prod_{K=0}^{n} Z_K \text{ to } s_j} \tag{12}$$

where the numerator is the product of the magnitude of the vectors from the poles to the point "s_j" and the denominator is the product of the magnitudes of the vectors from the zeros to the point "s_j".

The difficulty of obtaining the root loci is not a random process, however. The root loci are continuous curves, since the zeros of a polynomial are continuous functions of the coefficients. Certain rules and relationships permit rapid location of certain parts of the loci from which extensions can be made to determine other parts.(3)

Mechanical Aids. --At least two mechanical devices have been devised as aids in the construction of the root-locus. One of these, the Spirule, consists of a transparent protractor for the addition of the angles directly and a logarithmic spiral for multiplication of the

vector lengths to measure gain.(4) In a second, the Complex Plane Analyzer, the addition of angles and logarithmic magnitudes is accomplished electronically for a given pole-zero configuration and a selected point in the "s" plane.

Statement of the Problem. --The problem considered is divided into two parts. The first part was to obtain the transfer function of a given servo-mechanism using frequency-response techniques, to sketch the root-locus, and then to obtain the root loci using digital computing techniques. The second part of the problem was to arrange the computing techniques into a subroutine type of program in such a manner that alterations in the poles and zeros could be easily investigated.

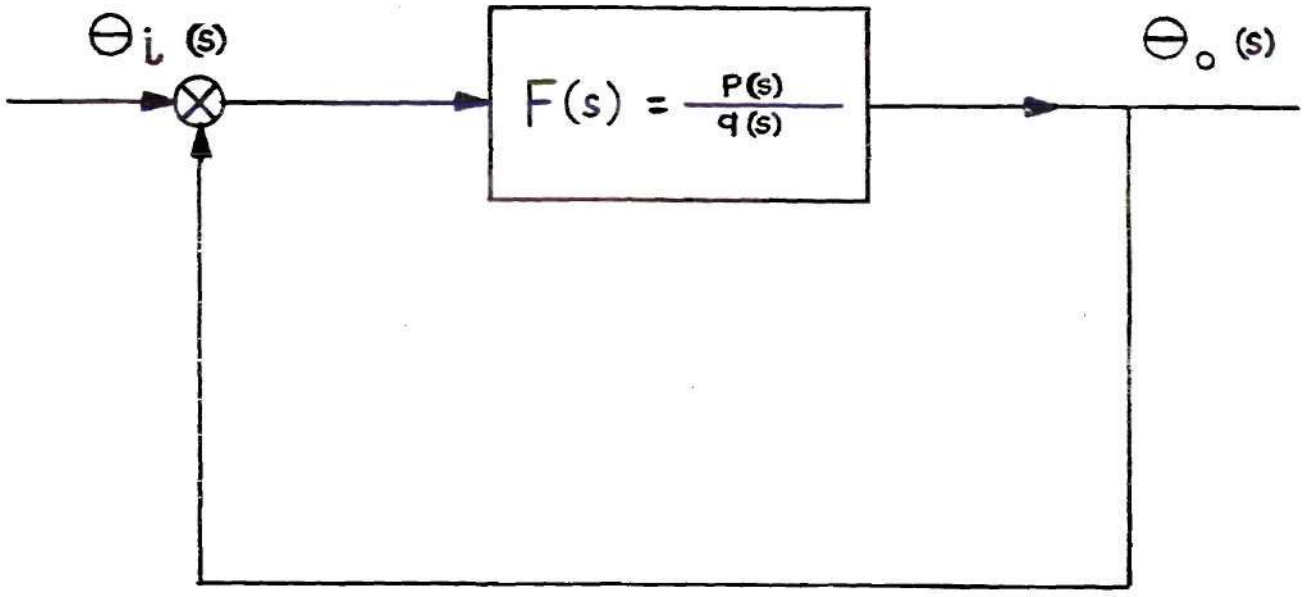

Figure 1. Single-Loop Feedback System.

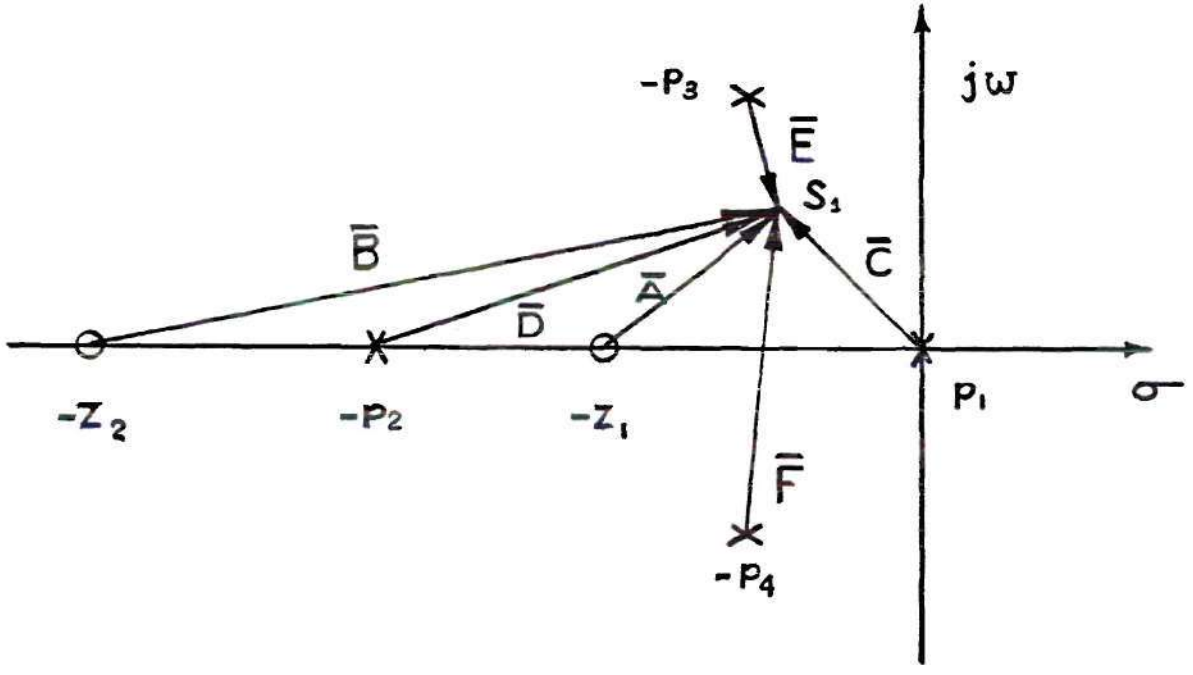

Figure 2. Graphical Interpretation of $F(s_1)$.

CHAPTER II

TRANSFER FUNCTION DETERMINATION

Method. --In order to apply the root-locus method in the analysis of a servo system it is first necessary to determine the open-loop transfer function for the system. This can be done in a number of ways. The most common are the use of transient and frequency-response techniques. If the system can be divided into separate components which do not interact then the transfer function of each component can be obtained by the above techniques. The open-loop system function is then the product of all the individual transfer functions.

The application of transient techniques to determine the transfer function of components is readily applied providing the transfer function to be determined can be represented by a single time constant. As the number of time constants increases the method becomes more difficult and time consuming. Since many physical devices do have a transfer function representation which is more complex than a single time constant, this method is seldom used and has given way to the steady state frequency-response method.

In the steady state frequency-response method, the output and input amplitude and phase are related as a function of frequency. By plotting the output-input amplitude ratio as a function of frequency it is

possible to use logarithmic approximations and determine the transfer function readily. Using this method with the approximations, the determination of complex transfer functions is only slightly more difficult than those of a single time constant. For this reason this approach was best suited in determining the transfer function of the servo system to be investigated.

The Uncompensated System. --The open-loop transfer function was desired for the system shown in Figure 3, page 12. This system is given in complete detail in Figures 32 and 33 in the Appendix, pages 98 and 99. Referring to Figure 32, page 98, switch No. 1 controls the compensation and was to remain open throughout the experimental procedure on the uncompensated system. Switch No. 2 determines whether the system operates with or without feedback.

The original approach was to determine the transfer function of each component, then, by multiplying these together, obtain the system transfer function. This approach is only correct when loading effects of successive components are small. In this particular system the assumption was not valid and a new approach was made toward determining the entire system transfer function as one unit. An open-loop system was used as shown in Figure 4, page 13, where the system was driven with a suppressed carrier signal from a Series 1100, Model A, Servoscope. The output obtained from the synchro control transformer was to be compared with the input from the Servoscope in the form of

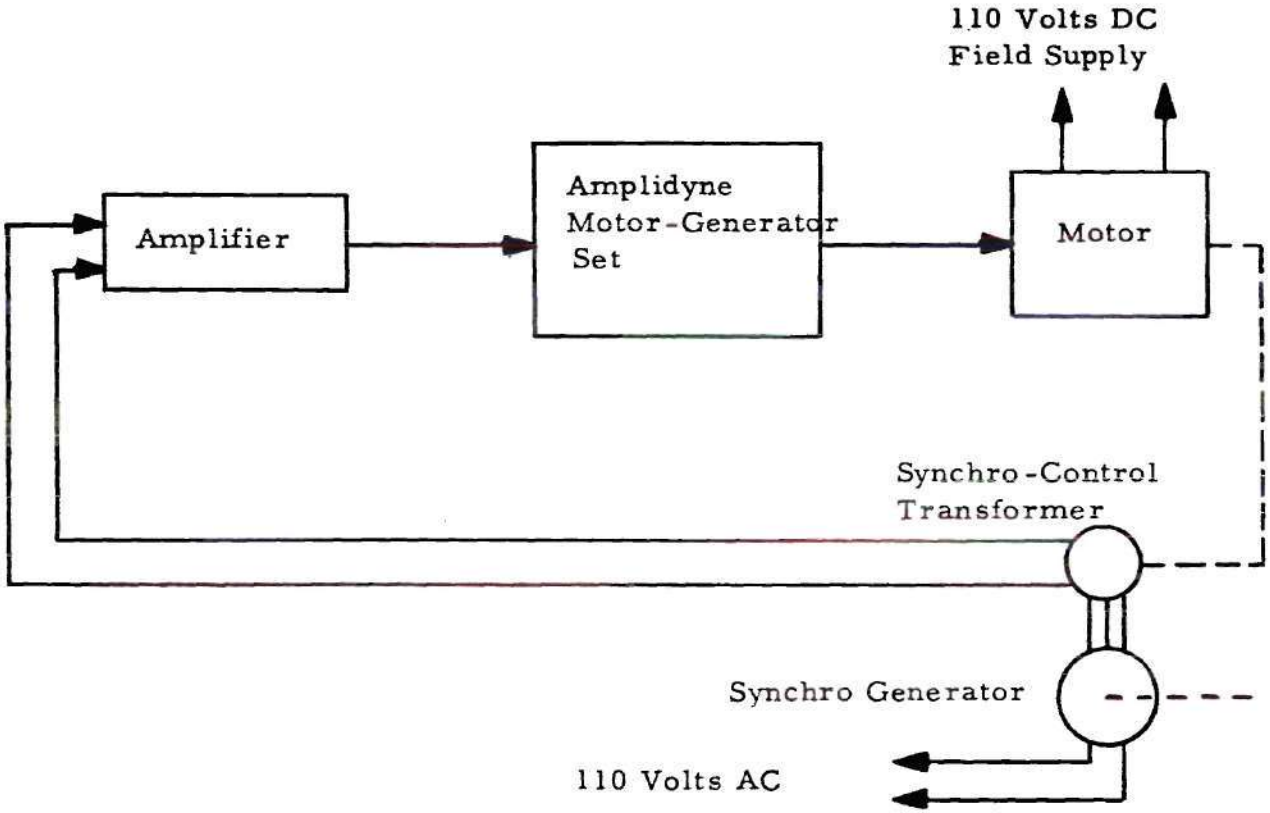

Figure 3. Feedback System to be Analyzed.

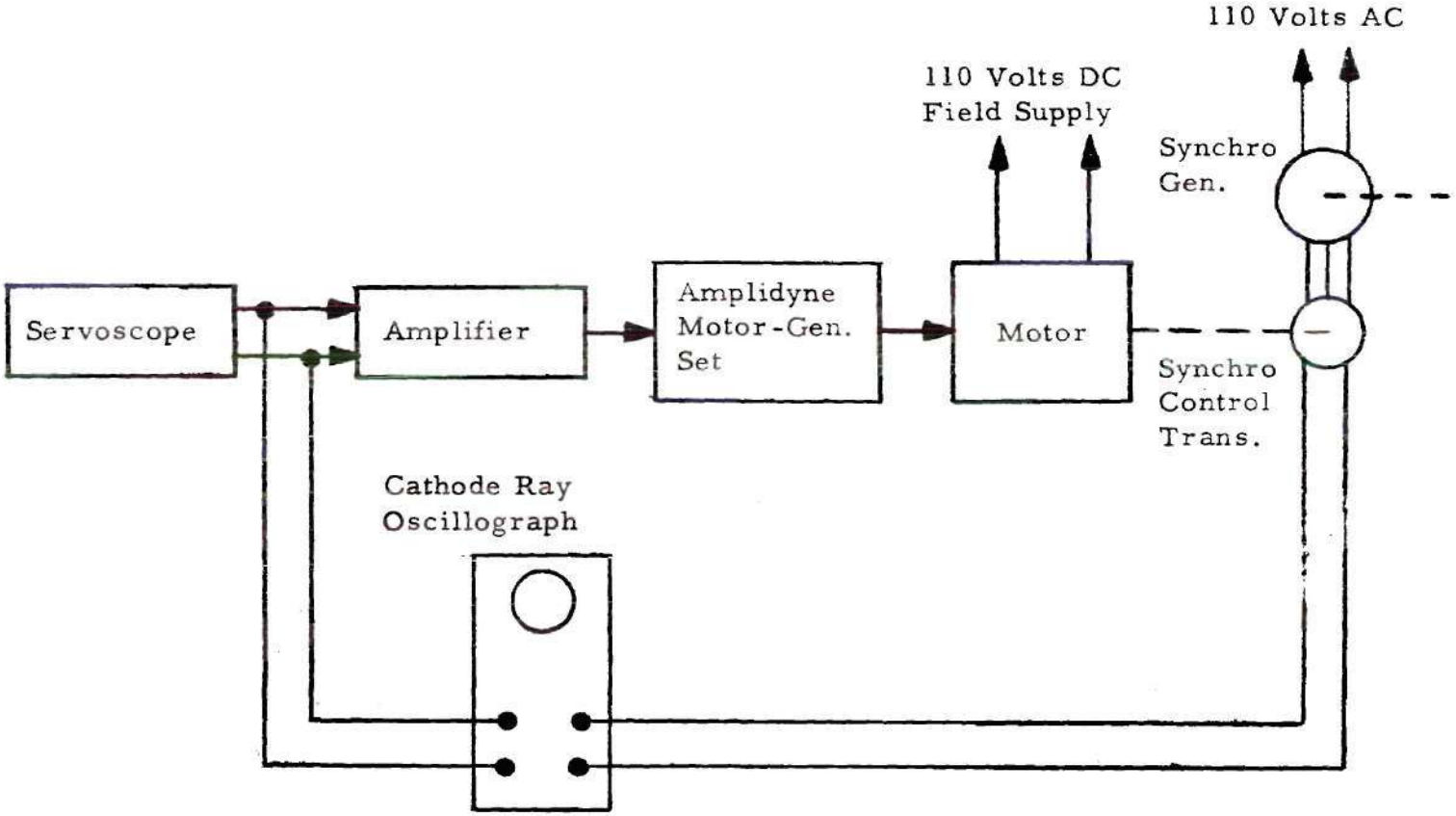

Figure 4. Measurement of Open-Loop Characteristics.

Lissajous patterns and relative amplitude measurements as the frequency of the input signal was varied. Under these conditions the system tended to drift from a zero reference. Unbalanced voltages in the system tended to drive the motor in an asymmetric manner which effectively caused the output from the control transformer to drift instead of maintaining an even oscillation about a fixed point. The drift caused the gain readings to be inaccurate, and measurements were difficult to obtain.

One way of overcoming these difficulties is to close the loop (switch number 2, Figure 32, page 98), thus stabilizing the system. If the open loop characteristics are to be measured this introduces new difficulties in measurement since the input signal is now the difference between two signals which is an incremental voltage for high gain loops. To enable accurate measurements to be made it was necessary to place a gain control potentiometer in the feedback path. The potiometer was set at a value which would provide sufficient feedback to remove the drift and enable accurate measurements to be made. The system was then functioning as a low gain closed loop system. The system shown in Figure 5, page 15, was the set-up actually used in obtaining the frequency-response of the system. The potentiometer of 15,000 ohms across the control transformer serves as the gain control for the system. With reference to the above explanation, this potiometer was set at 300 ohms during the experimental procedure. The amplitude of the

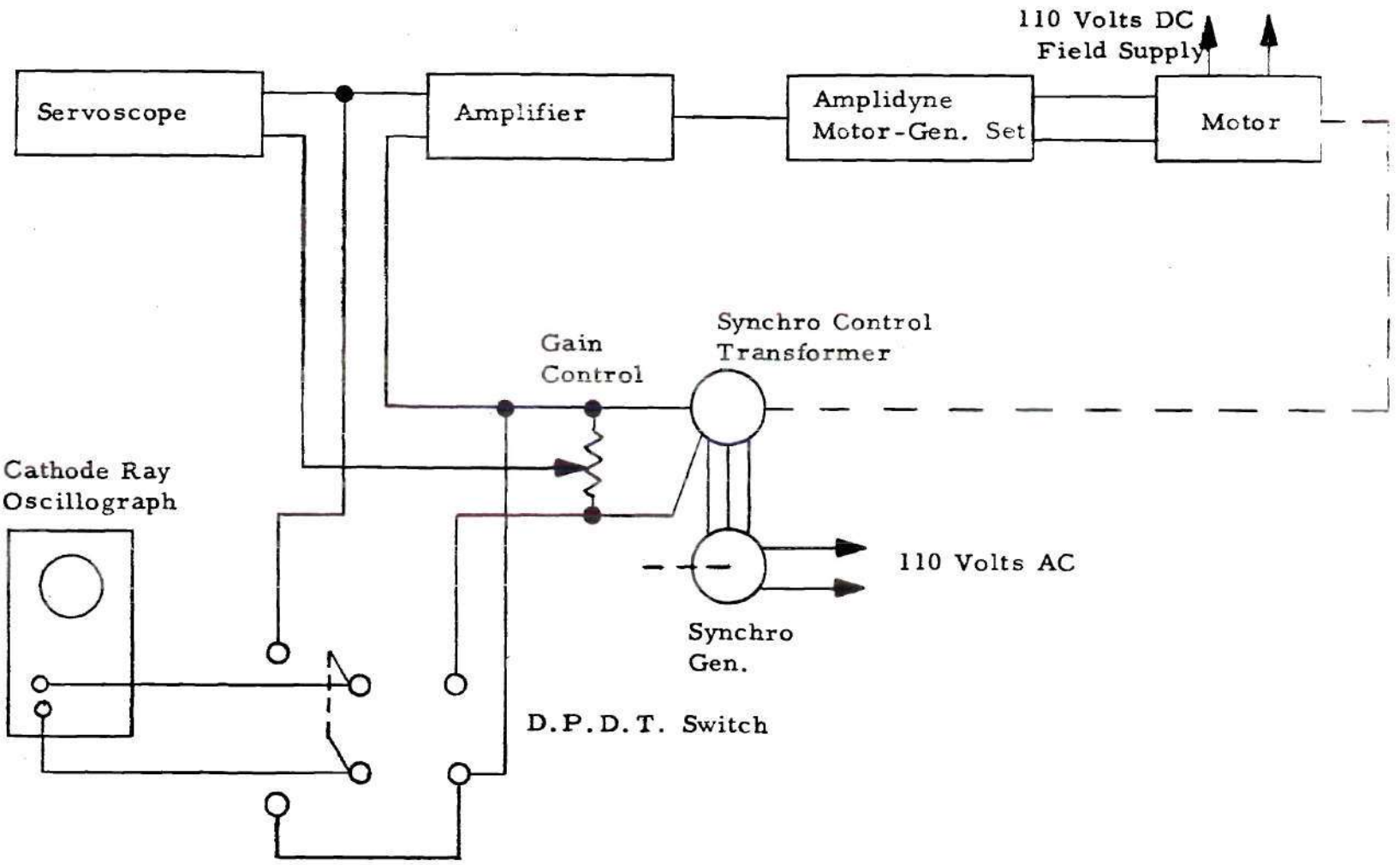

Figure 5. Open-Loop Attenuation Measurements Using a Closed-Loop System.

reference signal supplied by the Servoscope was 0.23 volts peak-to-peak. The relative amplitude of the input and output signal were measured on the cathode-ray oscillograph as the frequency of the reference input was varied. These values are given in Table 1, page 67, in the Appendix.

The gain in decibels , given by the relation,

$$\text{Gain in decibels} = K_{db} = 20 \log \frac{A_{out}}{A_{in}} \qquad (13)$$

was plotted versus the logarithm of the angular frequency. These calculations are also given in Table 1 and the values plotted on the curve shown as Figure 6, page 17. The asymptotic approximation was next constructed according to references (5), (6) and (7) in the Bibliography. The accuracy of this approximation is indicated by a comparison of the actual frequency response curve and the curve produced by the approximating function shown in Figure 7, page 18. The greatest deviation occurs near the low frequency end. It was known, however, that the dc motor should cause the system to have zero steady-state error, or in terms of the "s" plane, a pole at the origin. Hence the initial slope was started at minus six decibels per octave to agree with the physical interpretation, although the measurements did not actually indicate this. It was felt that at this frequency and amplitude of input signal the friction was of a sufficient magnitude to cause the system to behave in a non-linear fashion, thus making the results somewhat inaccurate when

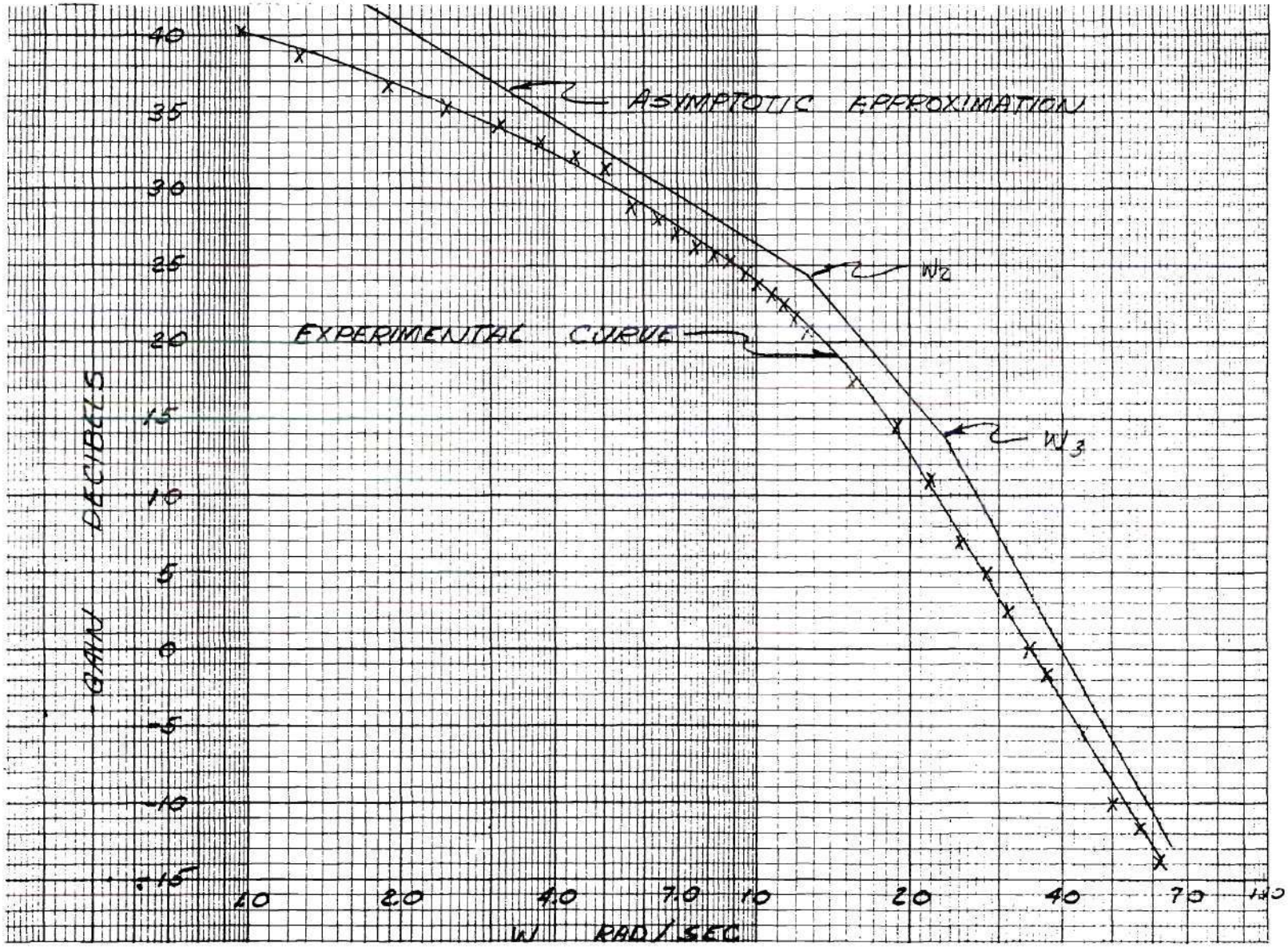

Figure 6. Open-Loop Frequency Response with Asymptotic Approximation.

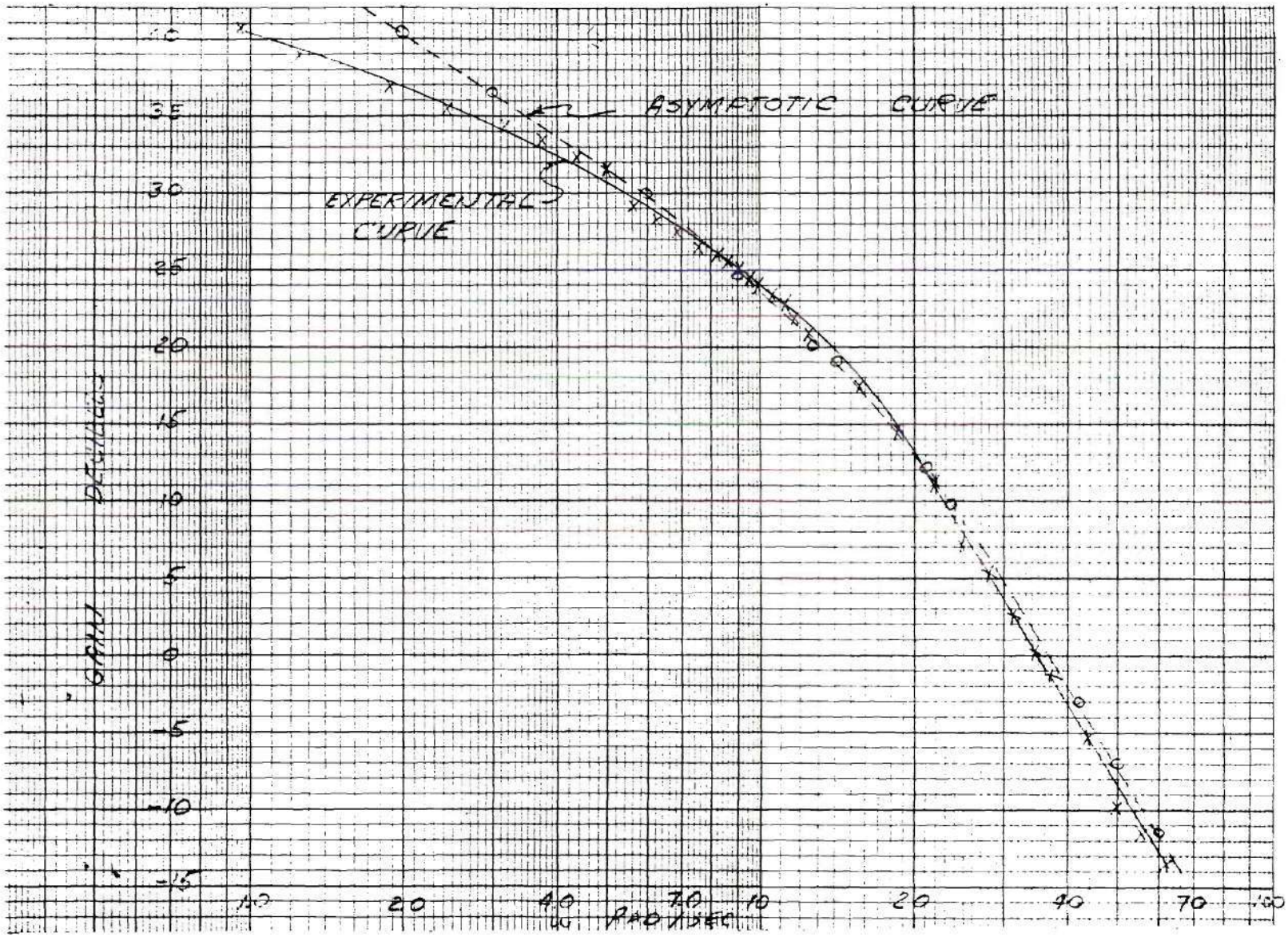

Figure 7. Comparison of Asymptotic Curve with Experimental Curve for Uncompensated System.

using linear techniques. Since agreement was consistent elsewhere, the physical interpretation of the system was relied on here and linear analysis was maintained.

The intersections of the various portions of the asymptotic approximations (see Figure 6, page 17) give the break or critical frequencies to be:

$$\omega_1 = 1/T_1 = 0.0 \tag{14}$$

$$\omega_2 = 1/T_2 = 12.6 \tag{15}$$

$$\omega_3 = 1/T_3 = 23.2 \tag{16}$$

From which the open-loop transfer function for the uncompensated system can be written as

$$\frac{\theta_o(s)}{\theta_i(s)} = \frac{K}{s(s + 12.6)(s + 23.2)(12.6)(23.2)} \tag{17}$$

The system has now been reduced to an analytic form and the methods of Chapter 1 can now be applied to this equation in terms of a root-locus system analysis. This work was followed by applying compensation and repeating the procedure.

The Compensated System. --The compensation to be used was obtained from the armature voltage of the motor. This voltage, which is approximately proportional to the first derivative of the output, was fed to a potentiometer so that the amount of feedback could be adjusted. This potentiometer voltage was used to drive the grids of the amplifier

(see Figure 33, page 99).

The experimental set-up was as shown in Figure 8, page 21. The gain control used was a 10,000 ohm potentiometer with the movable pick-off set at 300 ohms. The feedback potentiometer across the armature of the motor was 5,000 ohms total resistance with the variable arm set at 500 ohms. The amplitude of the reference input was 0.25 volts peak-to-peak. Measurements were made of the relative magnitude of the output and input signals as the frequency of the reference input was varied. The data and calculations similar to those for the uncompensated system are given in Table 2, page 68, of the Appendix. The frequency-response curve with its asymptotic approximation is shown in Figure 9, page 22. Figure 10, page 23, gives a comparison between the experimental curve and the curve derived from the approximating function. The logarithmic approximation shows that the compensation has added a zero and a pole to the system. The break frequencies (see Figure 9, page 22) obtained from the intersections are:

$$\omega_1 = 1/T_1 = 0.0 \tag{18}$$

$$\omega_2 = 1/T_2 = 4.42 \tag{19}$$

$$\omega_3 = 1/T_3 = 10.1 \tag{20}$$

$$\omega_4 = 1/T_4 = 12.9 \tag{21}$$

$$\omega_5 = 1/T_5 = 23.2 \tag{22}$$

From these, the open-loop transfer function for the compensated system

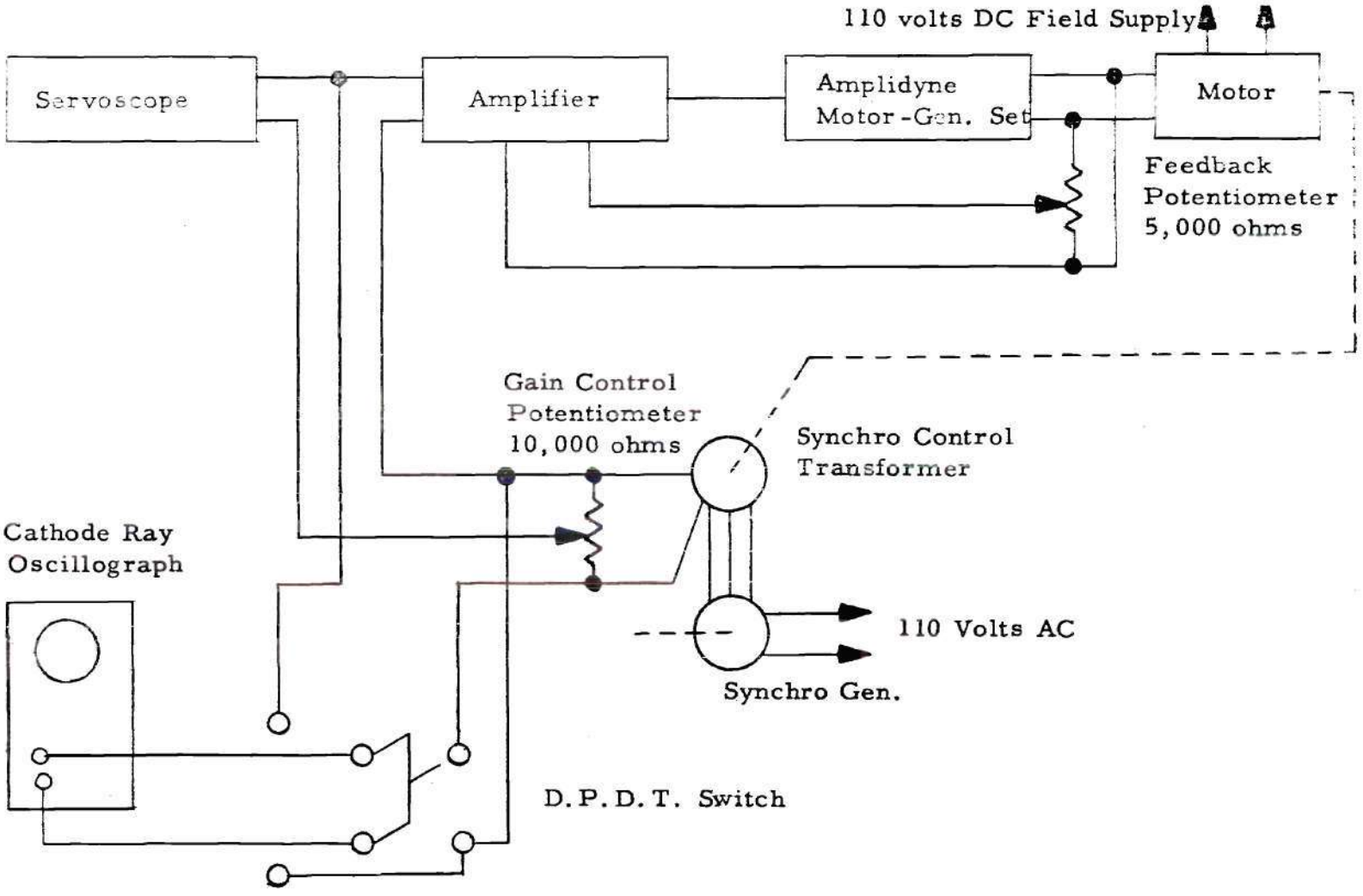

Figure 8. Measurement of Attenuation Characteristics for the Compensated System.

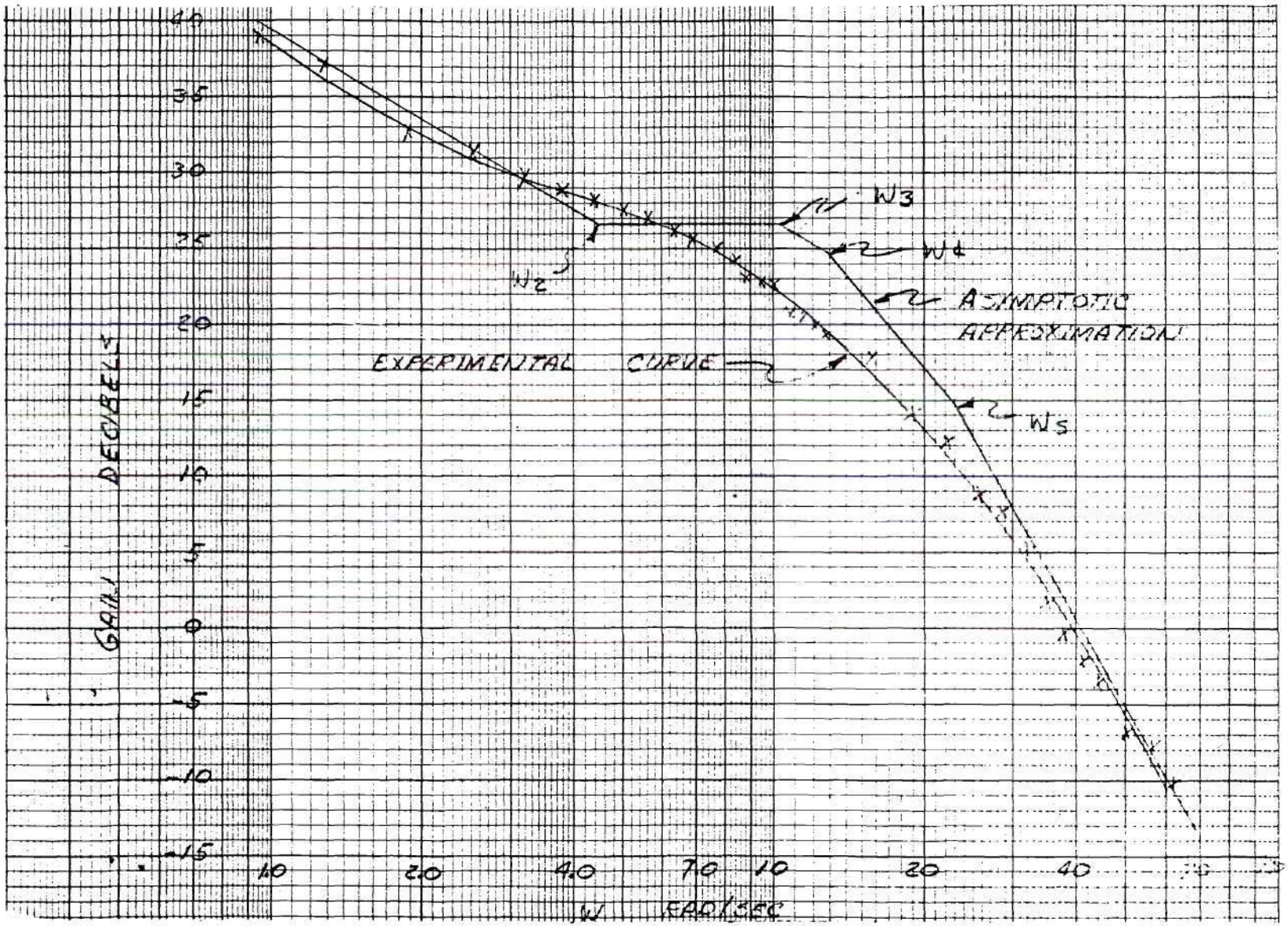

Figure 9. Open-Loop Frequency Response of Compensated System with Asymptotic Approximation.

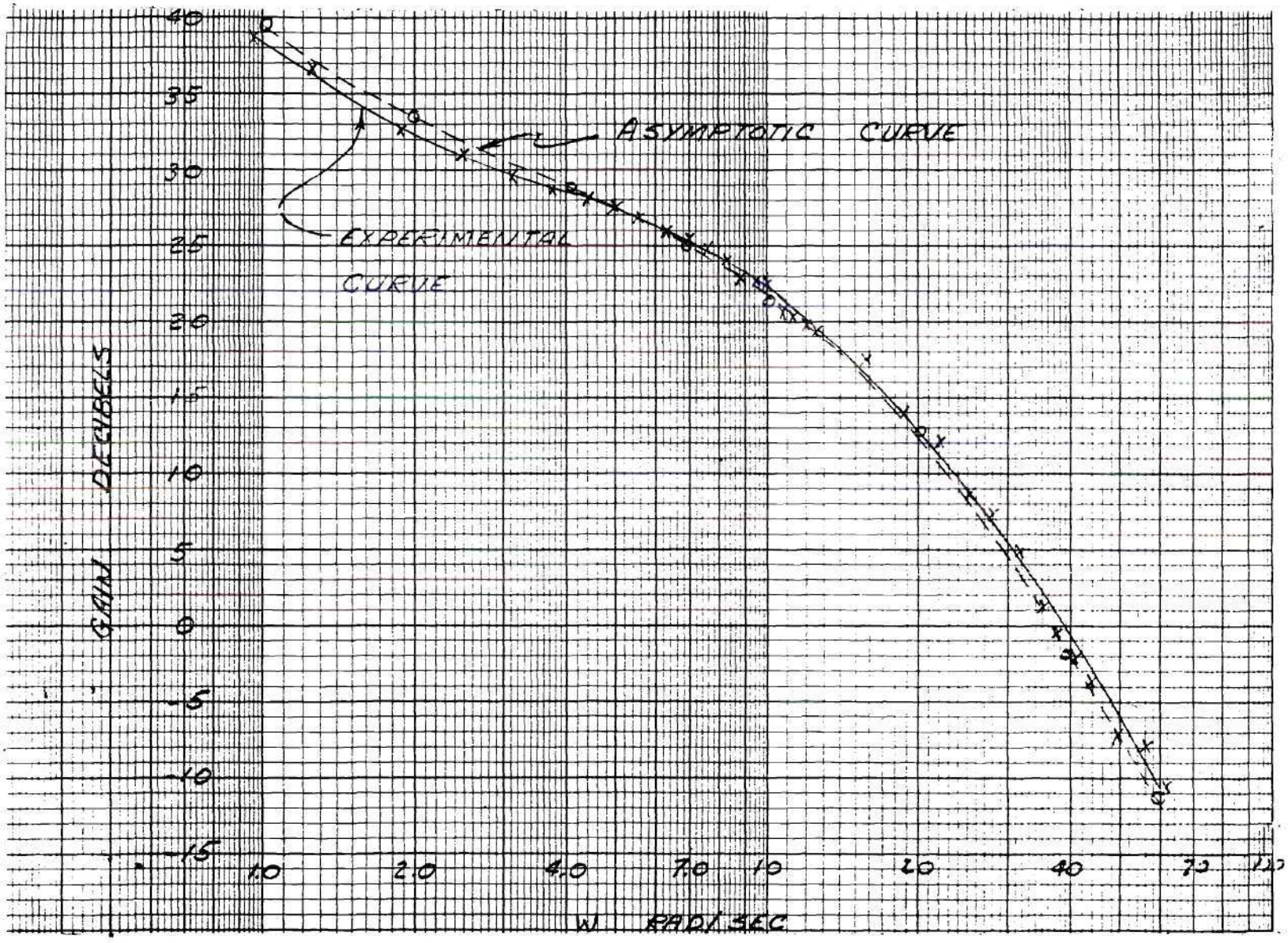

Figure 10. Comparison of Asymptotic Curve with Experimental Curve for Compensated System.

can be written as

$$\frac{\theta_o(s)}{\theta_i(s)} = \frac{K(s + 4.42)(4.42)}{s(s + 10.1)(s + 12.9)(s + 23.2)(10.1)(12.9)(23.2)} \tag{23}$$

Again the system has been reduced to an analytic form suitable for the application of root-locus analysis.

In the determination of the transfer functions in both cases, the attenuation characteristic alone was considered. This is not permissible for all problems, but in the case of physical equipment, generally this assumption can be made. (8) With the transfer functions thus determined the problem was then one of obtaining the root-locus for the system using the digital computer for the computations.

CHAPTER III

ROOT-LOCUS SOLUTION BY DIGITAL COMPUTER

Approximation For Loci.--The general concepts of the root-locus method have been stated in Chapter I. To use the method in the analysis of a servomechanism it is first necessary to obtain the open-loop transfer function as has been done in Chapter II. The next part of the analysis involves the computations of the root loci by application of the general equations (10) and (12), pages 5 and 6, to the specific open-loop transfer function. Since an approximation technique is required when using either equation it is desirable to have some method of obtaining a reasonable first guess. In solving for the root loci on a digital computer some ordered type of first approximation is necessary if computing time is to be held to a minimum.

The most logical answer to a first approximation to the root loci is a sketch based on the following special relations.*

(1) $K \rightarrow 0$. If the loci are interpreted as plots of the closed-loop pole positions when the open-loop gain varies from 0 to $+\infty$, the loci start at the open-loop poles.

(2) $K \rightarrow \infty$. When the open-loop gain varies from 0 to $+\infty$, the

---

*These special relations are based on a list of eleven given in the book Control System Synthesis by J.G. Truxal. New York: McGraw-Hill Book Company, 1955, pp. 227-234.

loci terminate on the open-loop zeros.

(3) Number of loci. The number of separate loci equals the number of poles or zeros of the open-loop transfer function, where critical frequencies at infinity are included and multiple-order critical frequencies are counted according to the order.

(4) Conjugate values. Complex parts of the loci always appear in conjugate complex pairs if, as is customary in the design of feedback systems, the coefficients of the polynomials p(s) and q(s) (see page 3) are real.

(5) Loci near infinity. As the "n" loci approach infinity, they tend toward asymptotes at angles of $\pm(180^{\circ} + m360^{\circ})/n$, where $m = 0, 1, 2, \cdots$,

(6) Point of breakaway from real axis. The point at which a locus breaks away from the real axis and becomes complex is defined as the point of breakaway. The equation for computing this point is derived in the Appendix, pages 65 - 66.

(7) Loci on real axis. Along the real axis, the angles of the vectors from any conjugate complex pair of zeros or poles of F(s) cancel, with the result that the total angle of $F(\sigma)$ is made up of the contributions from the real poles and zeros alone. At any given point, σ_1, on the real axis, the angle contributed by a pole or zero to the left of σ_1, is zero, while that from a pole or zero to the right of σ_1, is 180°. Hence the loci include those sections of the real axis to the left of an

odd number of open-loop critical frequencies.

With these special relations to work from, and others discussed in the reference, it is possible to make a rapid transition from the open-loop transfer function to the root-loci sketch. This seems a necessity in order to arrive at a reasonable program for a computer solution.

Solution for Uncompensated System. --The first step in the computer solution for the root loci was to obtain a general sketch using the rules stated in the preceding section. The sketch of the loci obtained from the open-loop transfer function for the uncompensated system (see equation 17, page 19) is as shown in Figure 11, page 29. The point of breakaway (preceding section-Rule 6), $-\alpha$, from the real axis is given by the equation (see Appendix, pp. 65-66).

$$\alpha = \frac{1}{3}(p_2 + p_3) - \sqrt{\frac{1}{9}(p_2 + p_3)^2 - \frac{1}{3}(p_2 p_3)} \tag{24}$$

The sections of the loci along the real axis were obtained from Rule 7 in the preceding section which made it necessary only to determine the gain at various points along the real axis loci. The symmetry of the loci, set forth in Rule 4 of the preceding section, reduced the computation problem to the upper half of the "s" plane.

Applying equation (10) to Figure 12, page 29, the computer problem to be solved was one of choosing an "s_1" such that

$$\phi_1 + \phi_2 + \phi_3 = \pi \tag{25}$$

where ϕ_1, ϕ_2 and ϕ_3 are the angles from the poles of the open-loop transfer function (equation (17), page 19) to the point "s_1" which lies in the "s" plane. Each "s_1", for which equation (25) is true, lies on the root-locus. The second problem was that of applying equation (12) and computing the gain at every value of "s_1" for which equation (25) was satisfied. Referring to Figure 12, page 29, the specific gain equation to be solved is

$$K = |\overline{A}|\,|\overline{B}|\,|\overline{C}| \tag{26}$$

where $|\overline{A}|$, $|\overline{B}|$, and $|\overline{C}|$ are the magnitudes of the vectors from the open-loop poles to the point "s_1" in the "s" plane. This equation can be written in terms of the system parameters by use of equations (14), (15) and (16) as

$$K = T_2 T_3 \left|\frac{1}{T_1} + s_1\right| \left|\frac{1}{T_2} + s_2\right| \left|\frac{1}{T_3} + s_1\right| \tag{27}$$

After a sketch of the loci was obtained, the value of α, or point of intersection with the real axis was first calculated. Increments were then added to α, at each of which an "s" was found such that equation (25) was satisfied, then K was found using equation (27).

The program was written using the General Purpose System for the International Business Machines 650 Digital Computer. This program is shown in the Appendix, pages 82-84. The flow chart, which

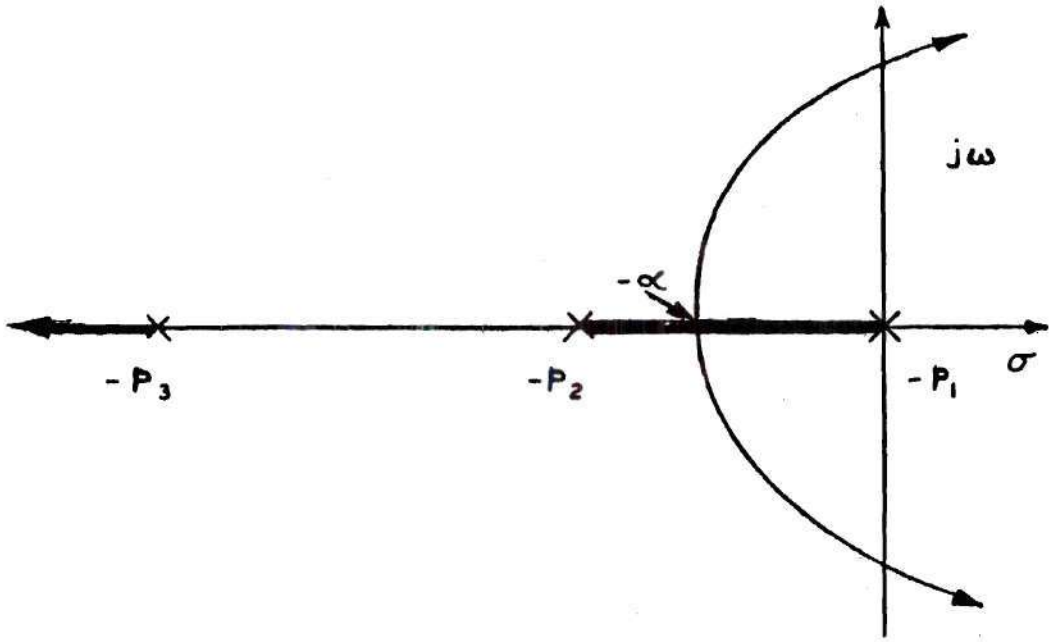

Figure 11 Sketch of Root-Locus for Three Poles.

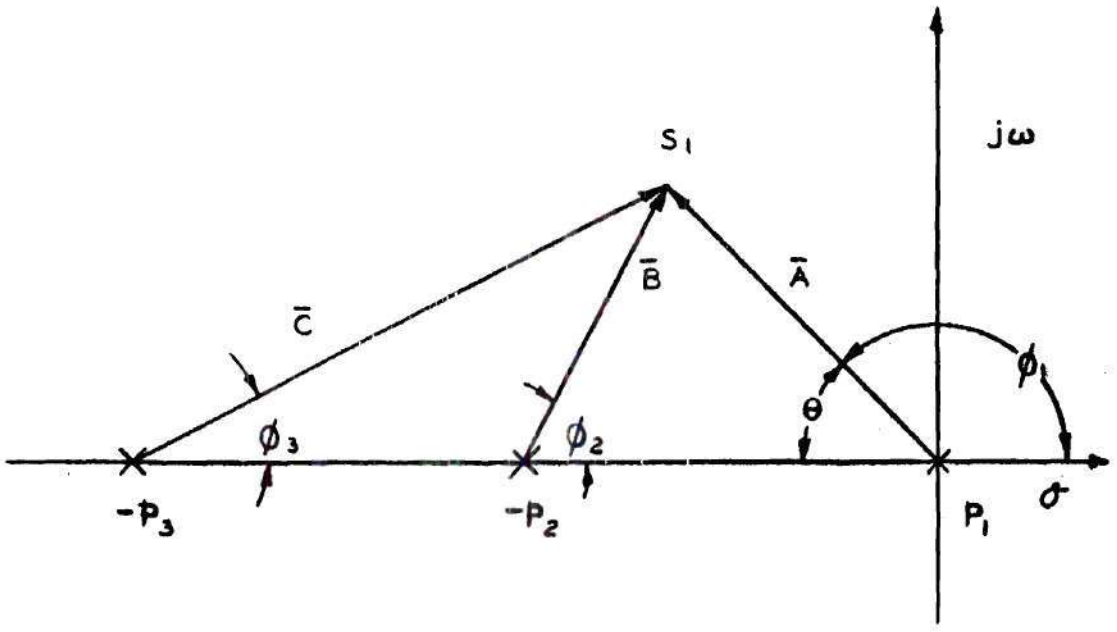

Figure 12. Graphical Interpretation of $\frac{\theta_o(s_1)}{\theta_i(s_1)}$

gives a graphical picture of the successive steps leading to the solution is shown in Figure 13, page 32. The box numbers given on the program in the Appendix refer to those shown in the flow chart, Figure 13. The steps in the solution will be explained with reference to Figure 13.

The open-loop poles and certain other constants were the only factors loaded into the memory drum for computation. This gives an indication of the adaptability of the program to a shift in the location of the poles. The approach to computation was to calculate the point of intersection with the real axis. On the basis of the sketch, the real problem then lay in determining values in the "s" plane which lay on the root-locus. Since it was known that the complex loci turned to the right, the problem was to find a method of converging on the complex loci as the point along the real axis was moved to the right in discrete steps.

In box 1 the point of intersection is computed according to equation (24). The answer is then punched out in box 2. The next symbol is program point 2, and as all program points, allows for re-entry into the program from some other point. This is necessary for successive iterations of the same procedure. Successive values along the real axis are computed in box 3 according to the equation

$$\sigma_{n+1} = \sigma_n + 0.500 \tag{28}$$

where $\sigma_0 = -\alpha$. Each time the program returns to program point 2, a new value of σ is determined giving successive values to the right, such

as $\sigma_1, \sigma_2, \cdots \sigma_n$, on successive iterations. The distances from the poles $-p_2$ and $-p_3$ to $\sigma_n (n = 0, 1, 2, \cdots)$ are computed in box 4. In box 5 successive values of $j\omega$ are obtained by adding a constant of 0.2 in this case, to the preceding value used, according to the equation

$$j\omega_n^{k+1} = j\omega_n^k + 0.200 \tag{29}$$

where $j\omega_1^0 = 0$. In general, the first value taken at any position, is given by

$$j\omega_{n+1}^0 = j\omega_n^m \tag{30}$$

where "m" represents the last of the iterations at the previous value calculated along the real axis. The notation $j\omega_n^k$ is to be interpreted as follows: The subscript "n" corresponds to that on σ_n and gives the correct pairing of σ and $j\omega$; the superscript "k" gives the number of the approximation which is being used at that particular value of σ_n. Hence for any value "n", the value "k" begins at zero and continues to increase until some prescribed limit of the error is reached.

Using this value of $j\omega$ the angles (see Figure 12, page 29) θ, ϕ_2, and ϕ_3 are computed if σ_n is negative, and the angles ϕ_1, ϕ_2 and ϕ_3 are computed if σ_n is positive. This discrimination is necessary in the computer solution due to a limited ability in handling transcendental functions. The angles ϕ_1, ϕ_2 and ϕ_3 always lie between 0° and 90°. However, θ will lie between -90° and 0°. Hence a test is

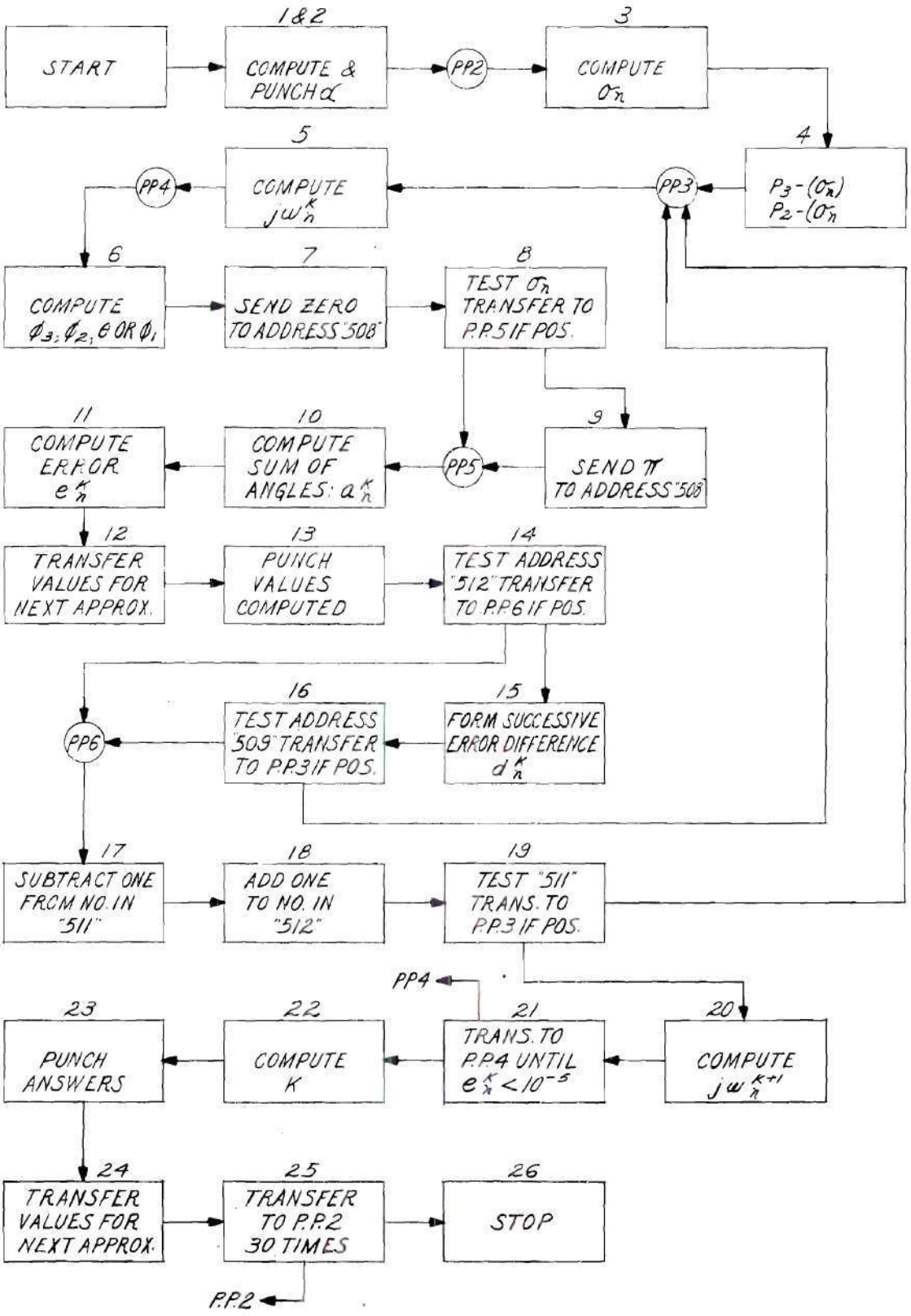

Figure 13. Flow Chart for Complex Solution of the Root-Locus for Three Poles.

necessary to determine which angle has been computed. In box 7 the value zero is sent to address "b" in preparation for the test to follow. The value σ_n is tested in box 8. If σ_n is negative the program goes to box 9 which sends the value π to address "b". Then in box 10 the sum

$$\phi_2 + \phi_3 + \pi - \theta = \phi_1 + \phi_2 + \phi_3 = a_n^{k+1} \tag{31}$$

is formed where a_n^{k+1} is the symbol for the sum of the angles at the real axis value σ_n and the imaginary axis value $j\omega_n^{k+1}$. If σ_n is positive, the sum

$$\phi_1 + \phi_2 + \phi_3 + 0 = a_n^{k+1} \tag{32}$$

is formed in box 10. Since the sum is supposed to equal the value π, the error is determined in box 11 by the relation

$$e_n^{k+1} = \pi - a_n^{k+1} \tag{33}$$

where e_n^{k+1} is the error which exists between the calculated value of $s = \sigma_n + j\omega_n^{k+1}$ and the root loci. In box 12 the values just computed are transferred in preparation for the next approximation and in box 13 the values $j\omega_n^k$, $j\omega_n^{k+1}$, e_n^k and e_n^{k+1} are punched out.

The error as a function of $j\omega_1^k$ is approximately as shown in Figure 14, page 36, for the first value, σ_1, computed along the real

axis. This is important in starting since approximating techniques tend toward the real axis, which also contains a value of "s" lying on the root-locus, instead of toward the complex value of "s" if sufficient care is not taken to insure that the maximum point has been passed. This is done in boxes 15 and 16. The difference is taken between successive errors, as

$$d_n^{k+1} = e_n^{k+1} - e_n^k \tag{34}$$

and as long as d_n^{k+1} is positive the program is repeated from box 5 to this point. However, when d_n^{k+1} becomes negative the right portion of the error curve has been entered and the program passes to box 17. In box 17 the value "one" is subtracted from "three" and the result stored in address "511", giving the numbers (2, 1, 0, -1, ...) in succession in this address. In box 19 address "511" is tested. If it is negative the program proceeds to box 20; if positive it transfers to program point 3 and repeats the calculations just discussed. This is designed to give three extra calculations after the maximum error point has been passed, to insure that the linear approximating technique which is to follow, is dependent upon the slope along the side of the curve and not some points near the maximum. With this precaution, linear approximation can be used very effectively in converging to the correct value of $j\omega$. In box 18 the value "one" is added to "minus two" and the result stored in address "512, giving the numbers (-2, -1, 0, 1,

...) in succession in this address. The number in address "512" is tested in box 14. Zero or positive values in this address enable the machine to by-pass boxes 15 and 16. This is a necessary precaution once the linear approximation is used to prohibit the program from returning to program point three.

In box 20, $j\omega_n^{k+1}$ is computed by the formula

$$j\omega_n^{k+1} = \frac{j\omega_n^k e_n^{k-1} - j\omega_n^{k-1} e_n^k}{e_n^{k-1} - e_n^k} \tag{35}$$

which is derived from a linear approximation of the error as shown in Figure 15, page 36. To be a valid approximation method, the values of $j\omega$ should converge to $j\omega_n^m$, or the error, e_n^k, should approach zero as "k" increases. The purpose of box 21 is to compare this error, e_n^k, with the nimber 1×10^{-5}. By the use of this test the program continues to transfer to program point four and repeat the calculations until the error is made smaller than 1×10^{-5}. Once the error is reduced below this value the program advances to box 22 and computes the gain at this particular "s" using the formula

$$K = \frac{\sqrt{(\sigma_n)^2 + (j\omega_n^m)^2}\sqrt{(p_2 - \sigma_n)^2 + (j\omega_n^m)^2}\sqrt{(p_3 - \sigma_n)^2 + (j\omega_n^m)^2}}{p_2 p_3} \tag{36}$$

In box 23 the values σ_n, $j\omega_n^k$, K, and e_n^{k-1} are punched out. In box 24, σ_n replaces σ_{n-1} and $j\omega_n^m$ replaces $j\omega_{n-1}^m$ in preparation

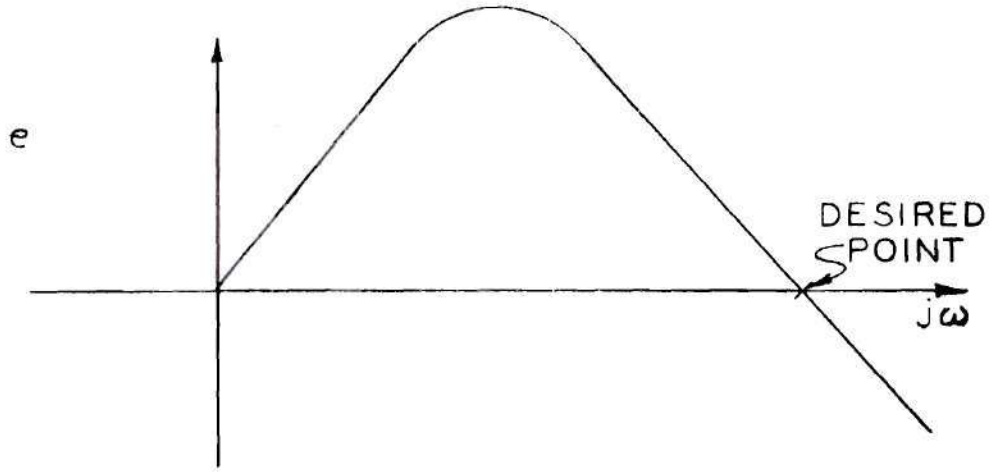

Figure 14. Error Curve for σ_1.

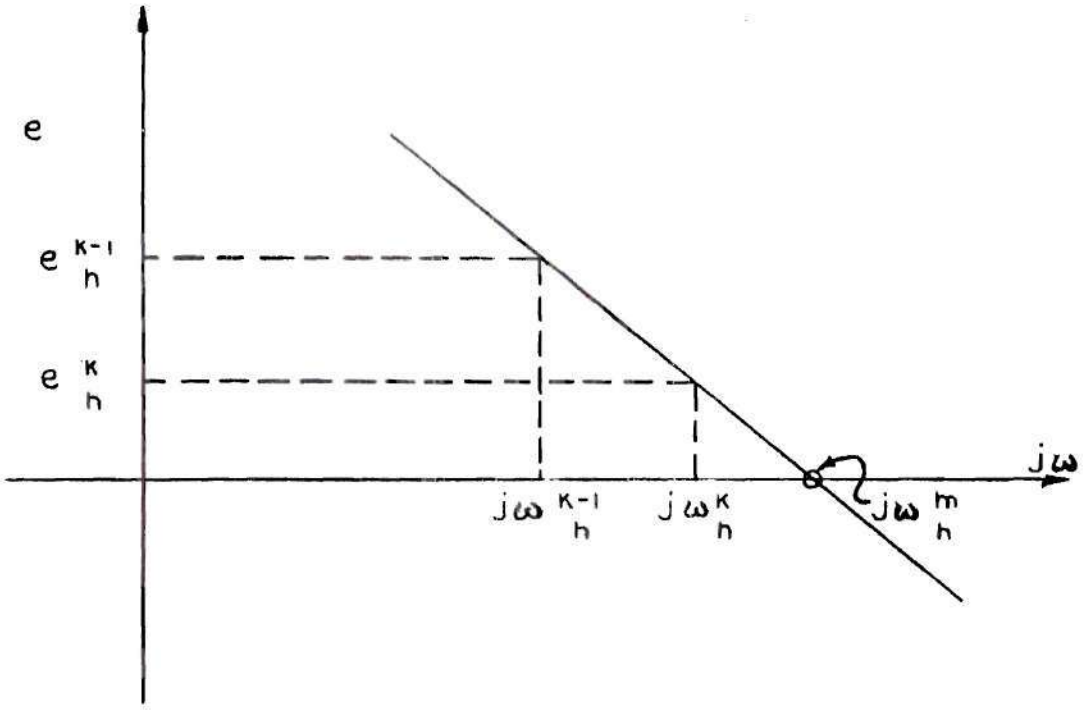

Figure 15. Linear Approximation of the Error Curve.

for a reiteration of the loop. The program transfers to program point two for 30 consecutive times, determined by box 25, after which the machine stops.

Table 3, page 69 in the Appendix, gives a list of the values punched out after they have been rounded off to three decimal places. The first value shown can be considered as σ_0, the point of break-away from the real axis. It should be emphasized that the increment along the real axis and the error limit were chosen arbitrarily and other values could be used just as well. Also, the number of points calculated could have been set at values other than 30.

An indication of the convergence is given in Table 5, page 71 in the Appendix. The first point shown is σ_1. It can be seen how the error increases to a maximum, proceeds with three more calculations in even steps after the negative sloping region is entered and then continues with the linear approximation method. Other points along the loci are also shown and indicate that four iterations were required to reduce the error to less than 1×10^{-5} in most of the cases. For most engineering work this magnitude of accuracy is not required, but the rapidity of convergence to any accuracy less than this can also be estimated from the table.

To obtain the values of gain along the real axis a second program was used. This could have been integrated into the other program, but to keep both as simple as possible it was done separately. The flow

diagram for this solution is shown in Figure 16, page 40. The program for the solution is given in the Appendix, page 85.

Referring to Figure 16 the program starts at zero and computes the gain for positive values, which are the same as those for negative values since magnitude is the only concern when computing gain. Equation (36) is used for this computation where $j\omega$ is equal to zero. The increment added each time is 0.5 from the pole p_1 to the pole p_2. When p_2 is reached the program transfers to the pole p_3, and continues calculating for 20 more points at intervals of 1.0 each. Again, both intervals and the number of calculations were arbitrary and other values could be used. The values obtained are given in Table 4, page 70 in the Appendix.

The root-locus for the uncompensated system was then plotted as shown in Figure 17, page 41, using Tables 3 and 4 in the Appendix. Values of gain are marked on the curve at every fourth point calculated. The plot indicates that the system is unstable for a value of gain greater than approximately 36 which is the point of intersection with the $j\omega$ axis. An appropriate compensation should modify the root-locus such that it intersects the $j\omega$ axis at a value of gain which is greater than that for the uncompensated system. The compensation used in this specific system and the resulting transfer function have already been discussed. An analysis of the system by the root-locus method will next be discussed where the root loci will be computed using a similar

approach to the method just discussed.

The Compensated System. --Applying the rules, for approximating the root-locus, to the transfer function for the compensated system (equation 23, page 24), a sketch of the root-locus can be determined as shown in Figure 18, page 43. From the shape of this curve it appears that the solution will not be greatly different from the uncompensated system with three poles. However, certain changes must be made in the program to find the point of breakaway from the real axis, and to determine what must be done as the locus passes across vertical lines erected from the critical frequencies.

The point of breakaway is no longer easily determined by a simple equation. Hence some type of approximation must be used. Referring to Figure 19, page 43, the equation to be used for the solution of the breakaway point can be determined. Choose a point "s_1", a distance "ϵ" above the real axis, and draw the vectors from the critical frequencies to "s_1", labeling the angles as shown. If "s_1" is to lie on the root-locus the following equation must be true:

$$\theta_1 - \phi_1 - \phi_2 - \phi_3 - \phi_4 = \pi + 2n\pi \tag{37}$$

where "n" is equal to any integer or zero. If "ϵ" is small, the tangents of the angles can be replaced by the angles as follows:

$$\tan (180^\circ - \phi_1) = \epsilon / \sigma_1 \cong 180^\circ - \phi_1 \tag{38}$$

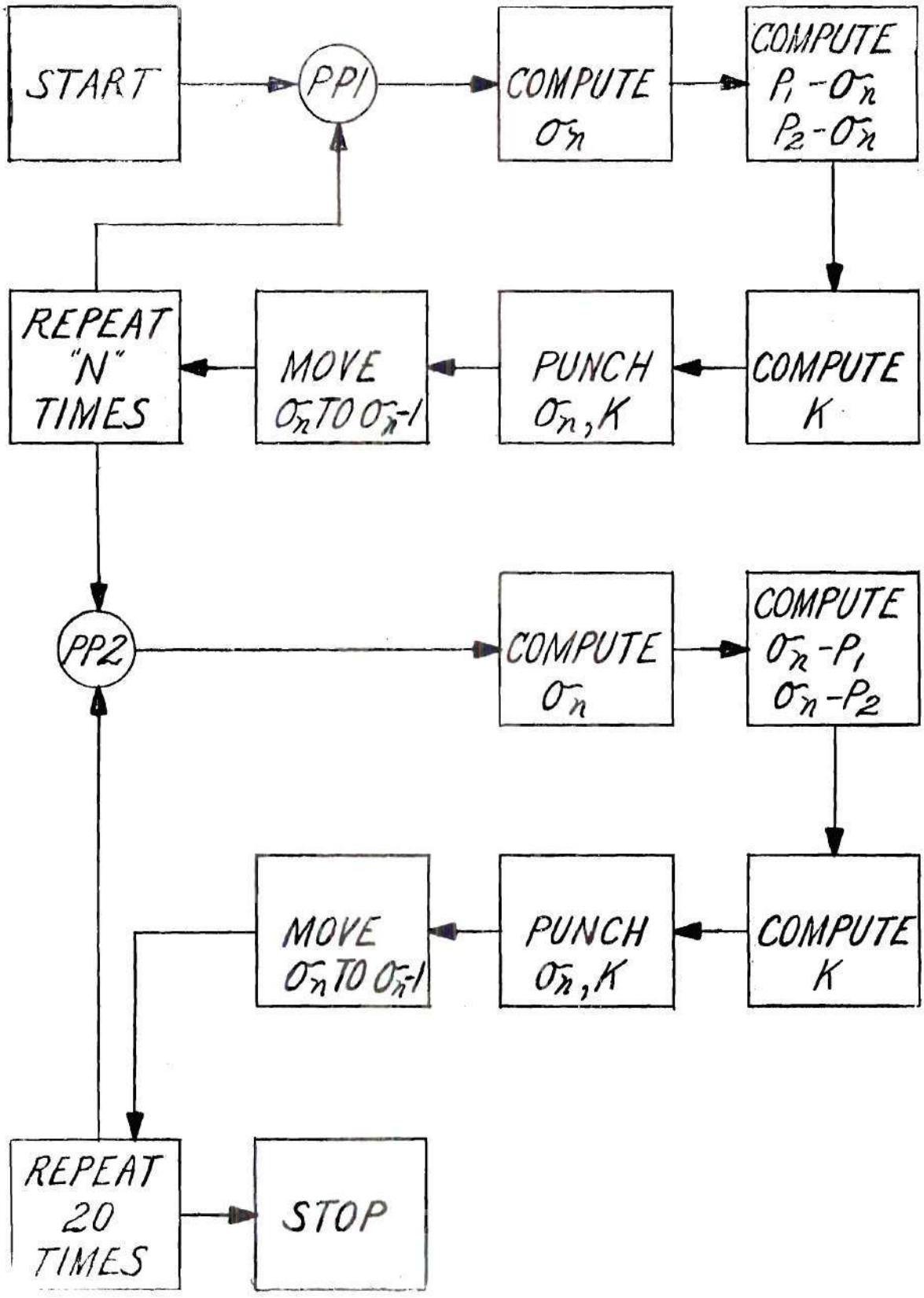

Figure 16. Flow Chart for Gain Solution Along the Real Axis.

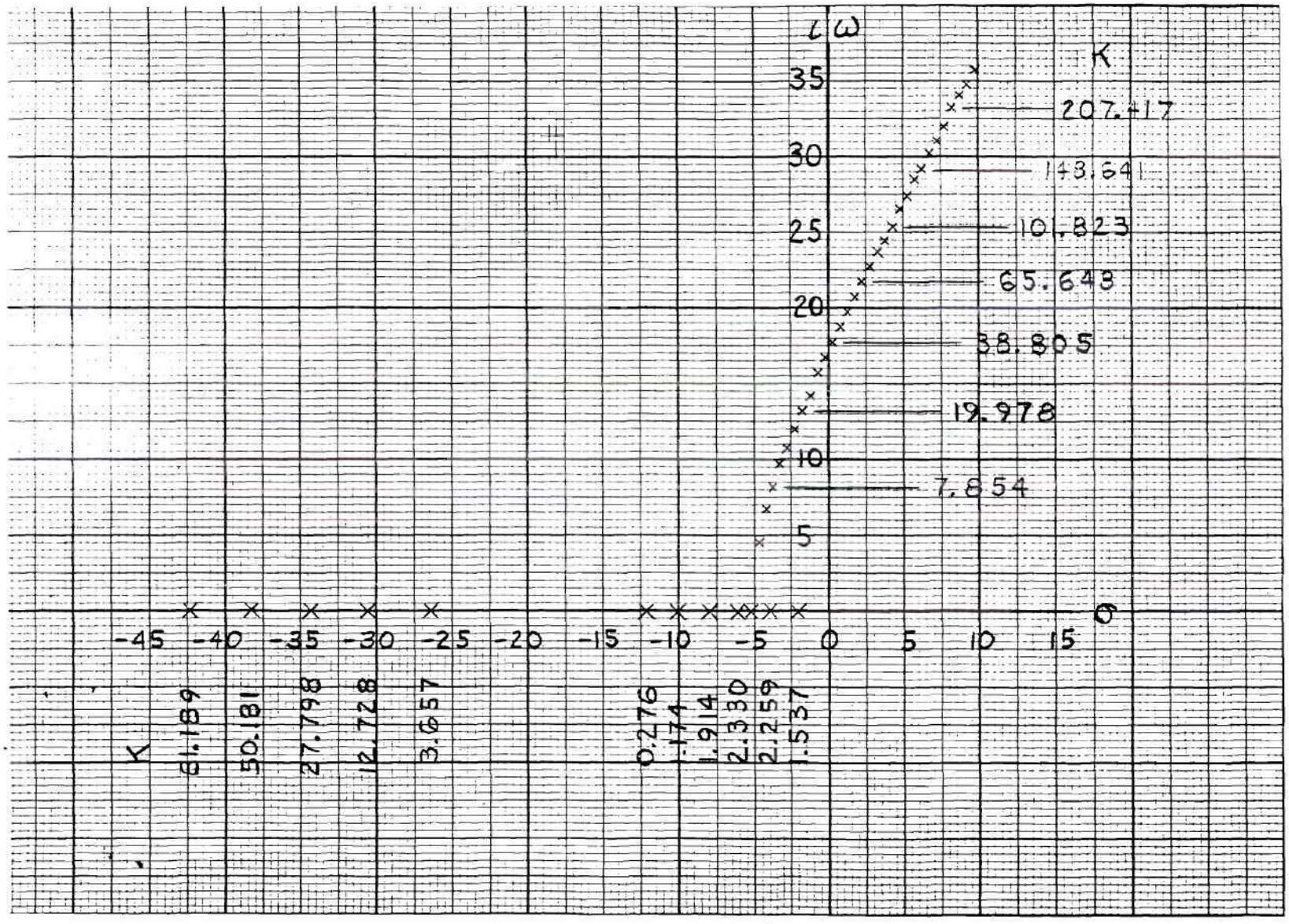

Figure 17. Uncompensated System Root-Locus Plot.

$$\tan(180^\circ - \phi_2) = \epsilon / \sigma_2 \cong 180^\circ - \phi_2 \tag{39}$$

$$\tan(180^\circ - \theta_1) = \epsilon / \sigma_5 \cong 180^\circ - \theta_1 \tag{40}$$

$$\tan \phi_3 = \epsilon / \sigma_3 \cong \phi_3 \tag{41}$$

$$\tan \phi_4 = \epsilon / \sigma_4 \cong \phi_4 \tag{42}$$

Substituting into equation (37), the equation to be solved is

$$\frac{1}{\sigma_1} \quad \frac{1}{\sigma_2} = \frac{1}{\sigma_3} \quad \frac{1}{\sigma_4} \quad \frac{1}{\sigma_5} \tag{43}$$

The method of solution can be seen in the flow chart of Figure 20, page 45. The first approximation, α_1, is taken half way between the pole p_2 and the pole p_3. The distances, their reciprocals, and the error, e_1, are then computed, where the error is given by

$$e_1 = \frac{1}{\alpha_1} + \frac{1}{\alpha_1 - p_1} = \left(\frac{1}{\alpha_1 - z_1} + \frac{1}{p_3 - \alpha_1} + \frac{1}{p_4 - \alpha_1} \right) \tag{44}$$

The second approximation, α_2, is equal to $\alpha_1 + 0.5$. The same calculations are made and a new error, e_2, obtained. With these values, the next approximation is obtained using the linear approximation formula:

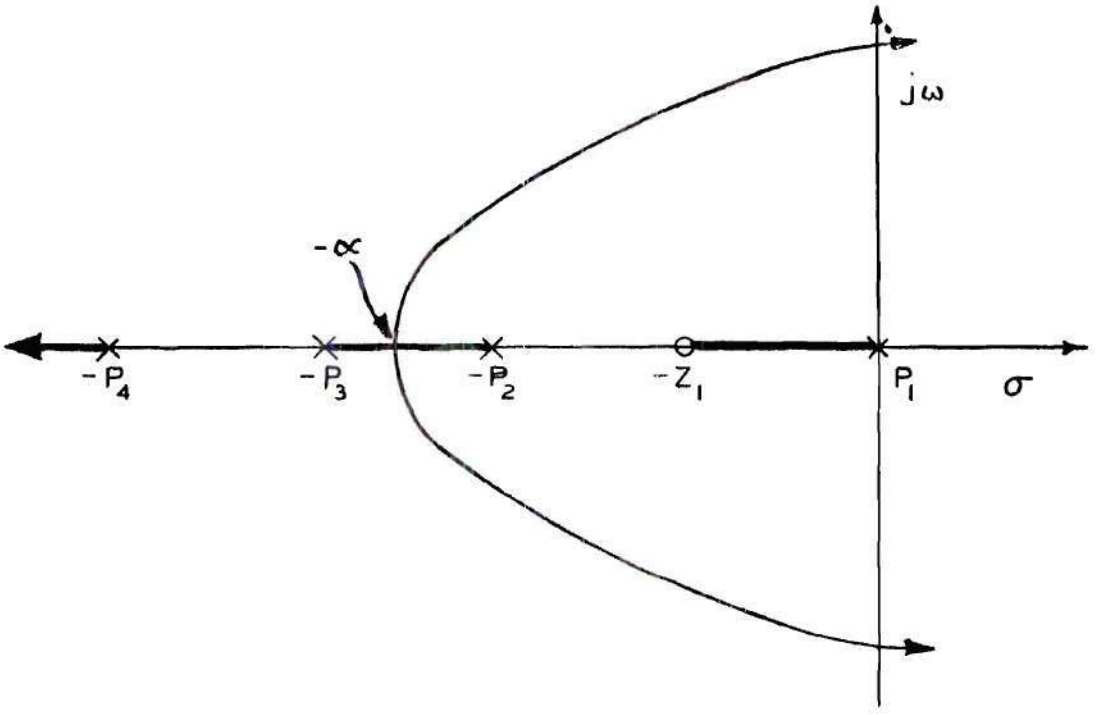

Figure 18 Sketch of Root-Locus for Four Poles and One Zero.

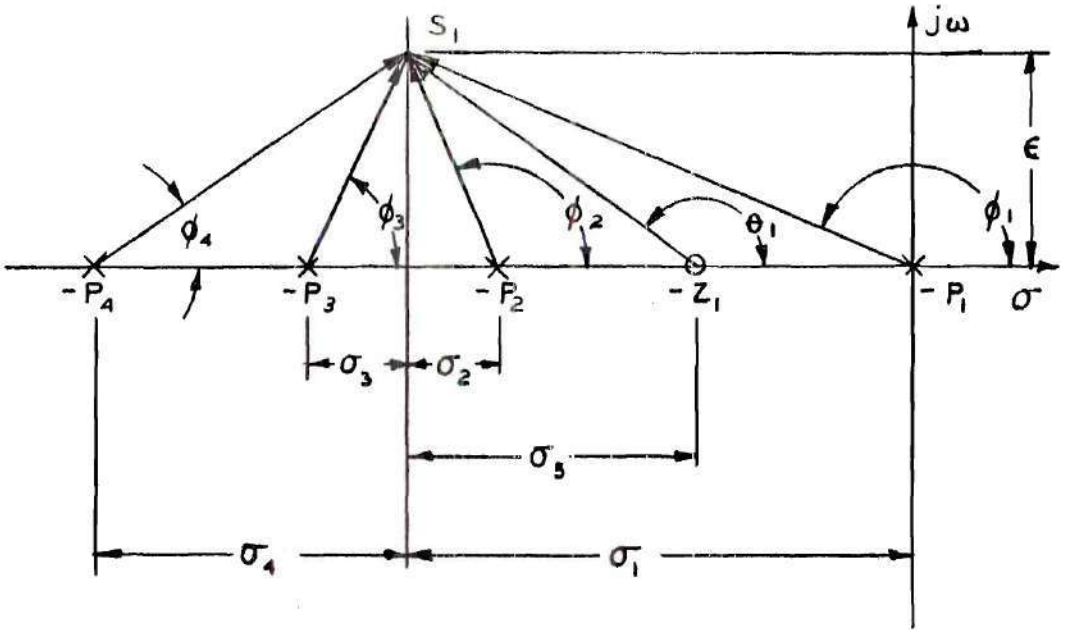

Figure 19. Determination of Breakaway Point.

$$\alpha_{k+1} = \frac{e_{k-1}\alpha_k - e_k\alpha_{k-1}}{e_{k-1} - e_k} \tag{45}$$

The error, e_k, is tested to see if it is less than 1×10^{-5}. If it is the machine stops or proceeds with the remainder of the program. If not, it moves e_{k-1} to e_{k-2}, e_k to e_{k-1}, α_k to α_{k-1} and α_{k+1} to α_k in preparation for the computation of α_{k+2}. The first part of Table 11, page 81 in the Appendix, shows the manner in which the method converges to the answer. Four iterations were necessary in order to reduce the error below the prescribed amount.

After the point of breakaway is determined the program begins the solution for the root-locus containing an imaginary part. Again, the sketch indicates that the problem is primarily one of determining the complex values of the root-locus. Much of this program given in the Appendix, page 86, is similar to the one previously discussed. Only those parts which are significantly different will be explained.

In the solution for the uncompensated system, the complex portion of the root-locus crossed only one vertical line erected from the critical frequencies. The sketch for the compensated system indicates that the complex portion of the root-locus will now cross three such lines. It can be seen in the program previously discussed that one of the major difficulties came at this crossover point where it was necessary to change the equations of system solution. From this it is also evident

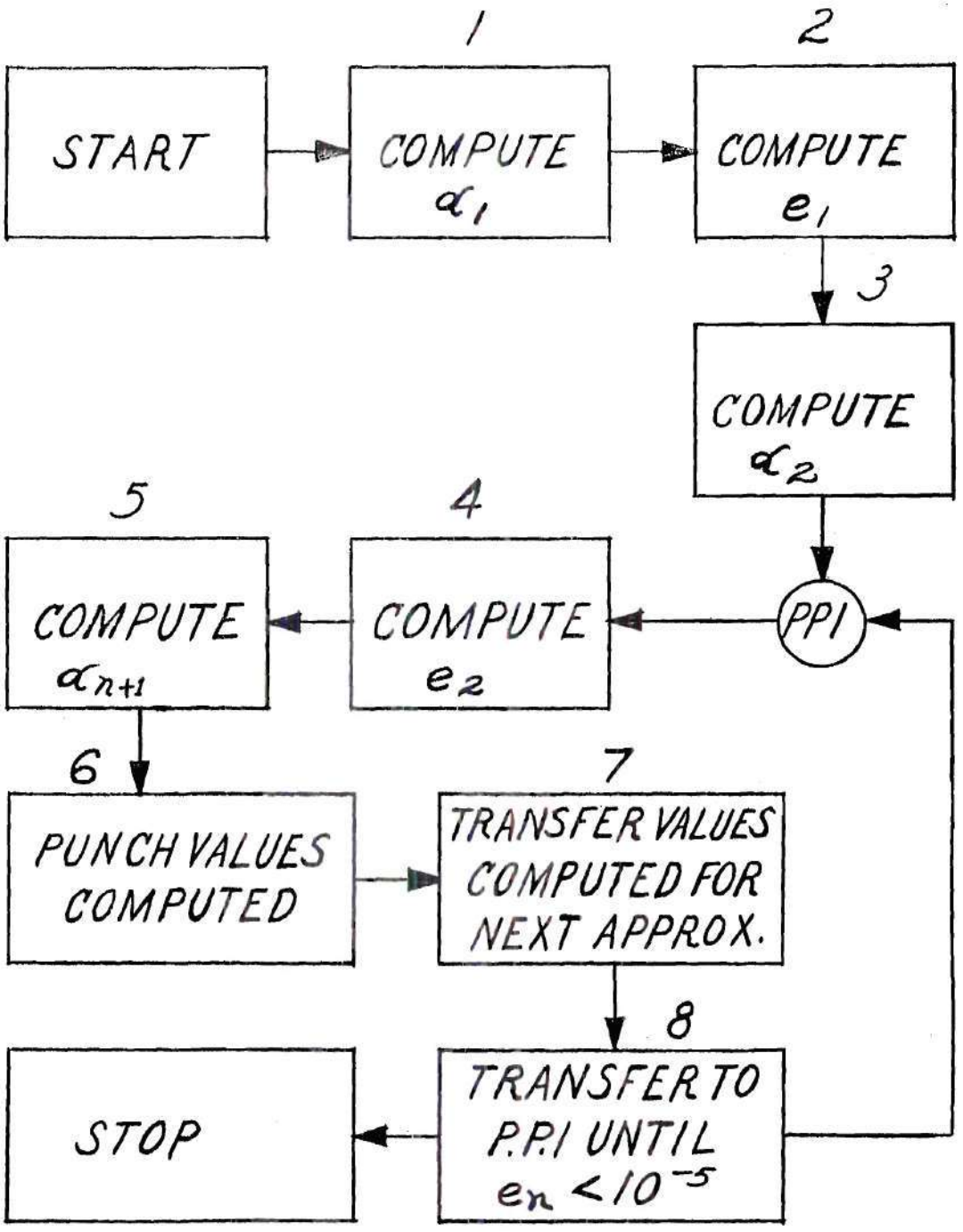

Figure 20. Flow Chart for Solution of Breakaway Point Four Poles - One Zero.

that a similar, but more complicated solution, is necessary for the compensated system.

The flow chart for the solution of this problem is shown in Figure 21, page 48. If the "s" plane is divided into the regions shown in Figure 22, page 49, then the angles which are computed in box 5, for various points on the root-locus are as follows: region 1-($-\phi_4$, ϕ_3, $\phi_2 - \pi$, $\phi_1 - \pi$, $\theta_1 - \pi$); region 2-($-\phi_4$, ϕ_3, ϕ_2, $\phi_1 - \pi$, $\theta_1 - \pi$); region 3-($-\phi_4$, ϕ_3, ϕ_2, θ_1, $\phi_1 - \pi$); and in region 4-(ϕ_4, ϕ_3, ϕ_2, ϕ_1, θ_1); where the angles to any point are as indicated in Figure 22, page 49. Equation (37) is the equation which must be satisfied in each instance. Then upon substitution, the equation which must be solved in each region is:

$$\text{Region 1:} \quad (\theta_1 - \pi) - \left[(\phi_1 - \pi) + (\phi_2 - \pi) + \phi_3 + \phi_4\right] - \pi = -\pi \tag{46}$$

$$\text{Region 2:} \quad (\theta_1 - \pi) - \left[(\phi_1 - \pi) + \phi_3 + \phi_4\right] = -\pi \tag{47}$$

$$\text{Region 3:} \quad \theta_1 - \left[(\phi_1 - \pi) + \phi_2 + \phi_3 + \phi_4\right] - \pi = -\pi \tag{48}$$

$$\text{Region 4:} \quad \theta_1 - \left[\phi_1 + \phi_2 + \phi_3 + \phi_4\right] = -\pi \tag{49}$$

Hence if the angles computed from the poles to the point in question are subtracted from the angle computed from the zero to the point, then the value "π" must be subtracted in regions 1 and 3 to keep the equations similar in form. This requires a test procedure to find the particular region in which the point is located and then procedure which follows the particular equation applicable.

The flow diagram for this particular decision circuit is shown in

Figure 23, page 51. First, the quantity $\left[-(p_2 - \sigma_n)\right]$ is tested. If this quantity is positive then the point is in region 1 and the program transfers to program point 5, proceeds to send the value "π" to address "b" and then goes to program point 6. If the number tested is negative the program continues and next tests the quantity $\left[-(z_1 - \sigma_n)\right]$. If this quantity is positive the point lies in region 2 and the program transfers to program point 6 and continues. If the quantity is negative the program continues and next tests the quantity $(p_1 - \sigma_n)$. If this quantity is negative the point lies in region 3 and the program transfers to program point 5, sends the value "π" to address "b" and then goes to program point 6. If the value tested is positive the point lies in region 4 and the program transfers to program point 6. After program point 6, the program is almost identical to the one for the uncompensated system. The sum of the angles is formed and the value in address "b" is subtracted. In case the point lies in regions 2 or 4, a zero which has been sent to address "b" is subtracted. In the other two regions the value "π" is subtracted, thus fulfilling the requirements of the various regions.

Values for the upper half of the "s" plane are given in Table 6, page 73 in the Appendix, after they have been rounded off to three decimal places. Again, the increment along the real axis, the error limit, and the number of calculations were arbitrary and could be changed.

Table 8, page 76 in the Appendix, gives an indication of the

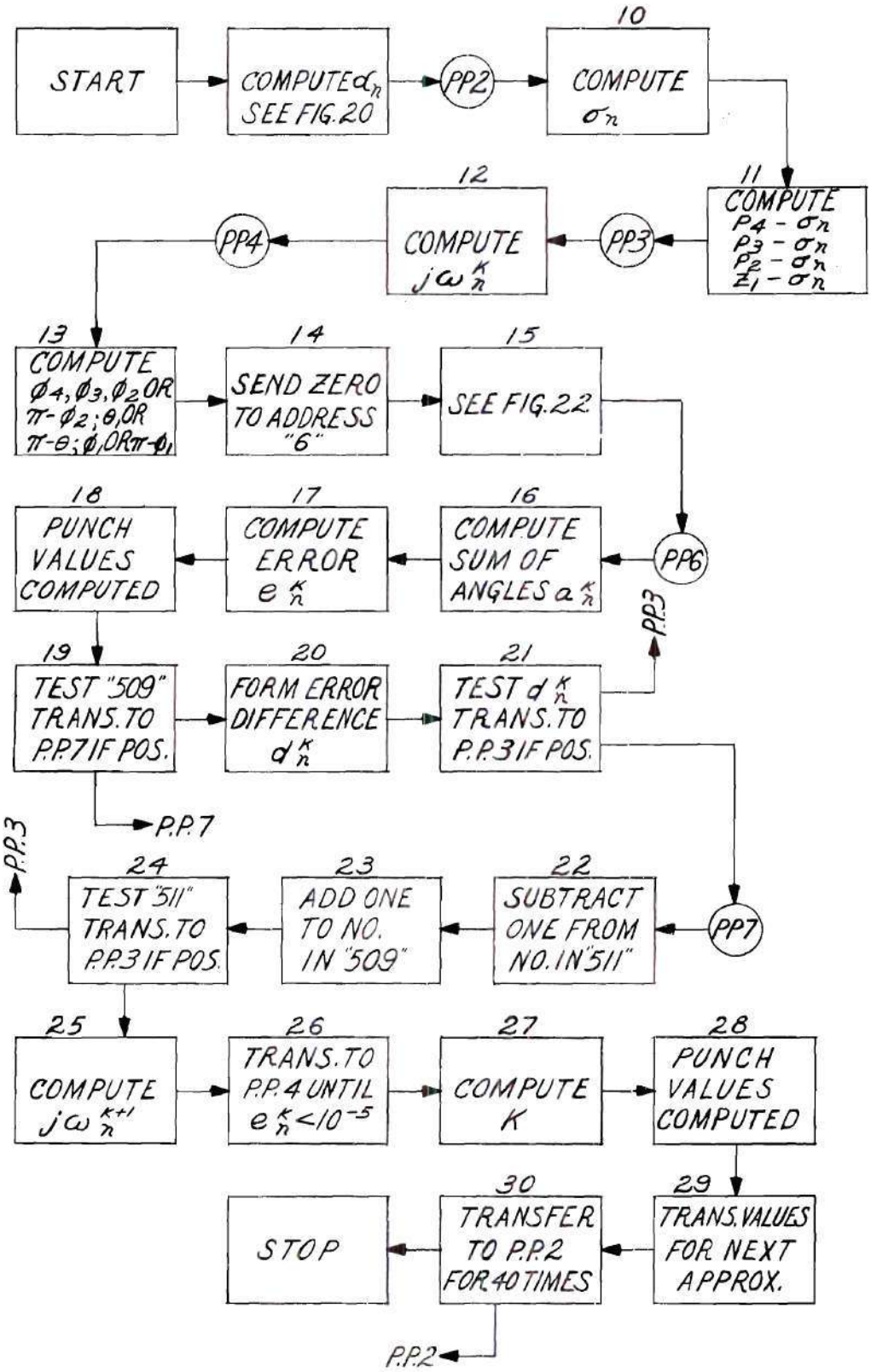

Figure 21. Flow Chart for Complex Solution of Root-Locus for Four Poles - One Zero.

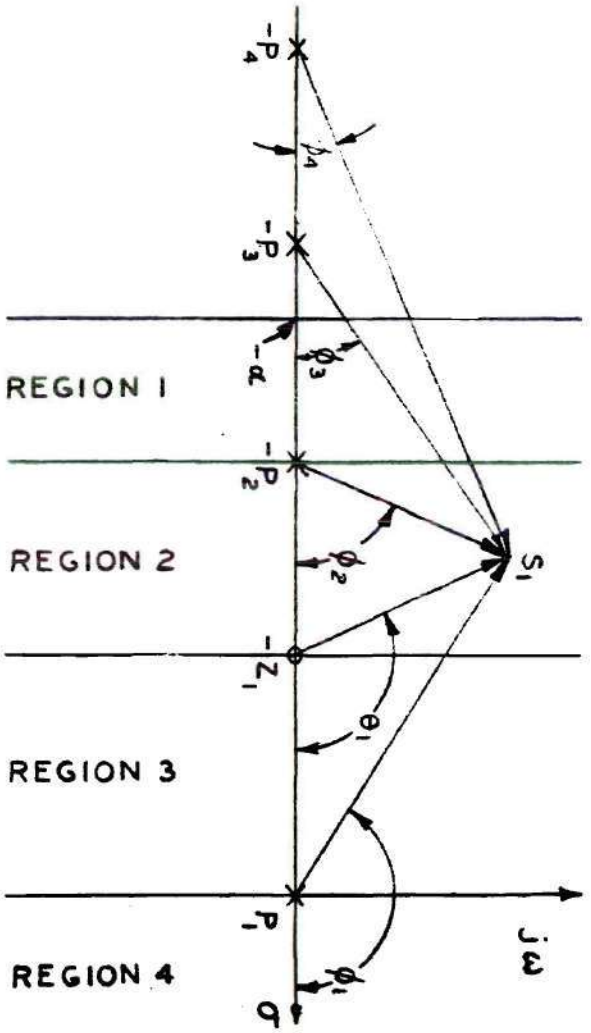

Figure 22. Regions of Calculation for Four Pole - One Zero Configuration.

convergence of the approximation method at various points along the root-locus. The first value, σ_1, has the same characteristics as for the three pole solution, since the method of solution is essentially the same. After this point, the table indicates that approximately four iterations were necessary to reduce the error to less than 1×10^{-5}.

The values of gain along the axis were computed by a separate program. Again this was merely a choice to keep the solution as simple as possible. The flow diagram, Figure 24, page 53, for the program given in the Appendix, page 91, indicates that this solution is also similar to the real axis solution for the three pole case. From p_1 to z_1 and from p_2 to p_3 increments of 0.2 were used. Beyond p_4 values were computed at unit intervals for 20 points. These values are shown in Table 7, page 75 in the Appendix.

The root-locus plot for the compensated system is shown in Figure 25, page 54 with the values of gain marked at every fourth point calculated. It can be seen that the system is unstable for values of gain greater than approximately 25, which indicates that the system response has been reduced. When system measurements were made in Chapter II an attenuator was placed in the feedback path. The negative result obtained using the compensation may be attributable to this attenuator, since the compensation was improperly placed for maximum effect, and magnitude too small for improving system performance. A better response could have been obtained by using a "parallel-T"

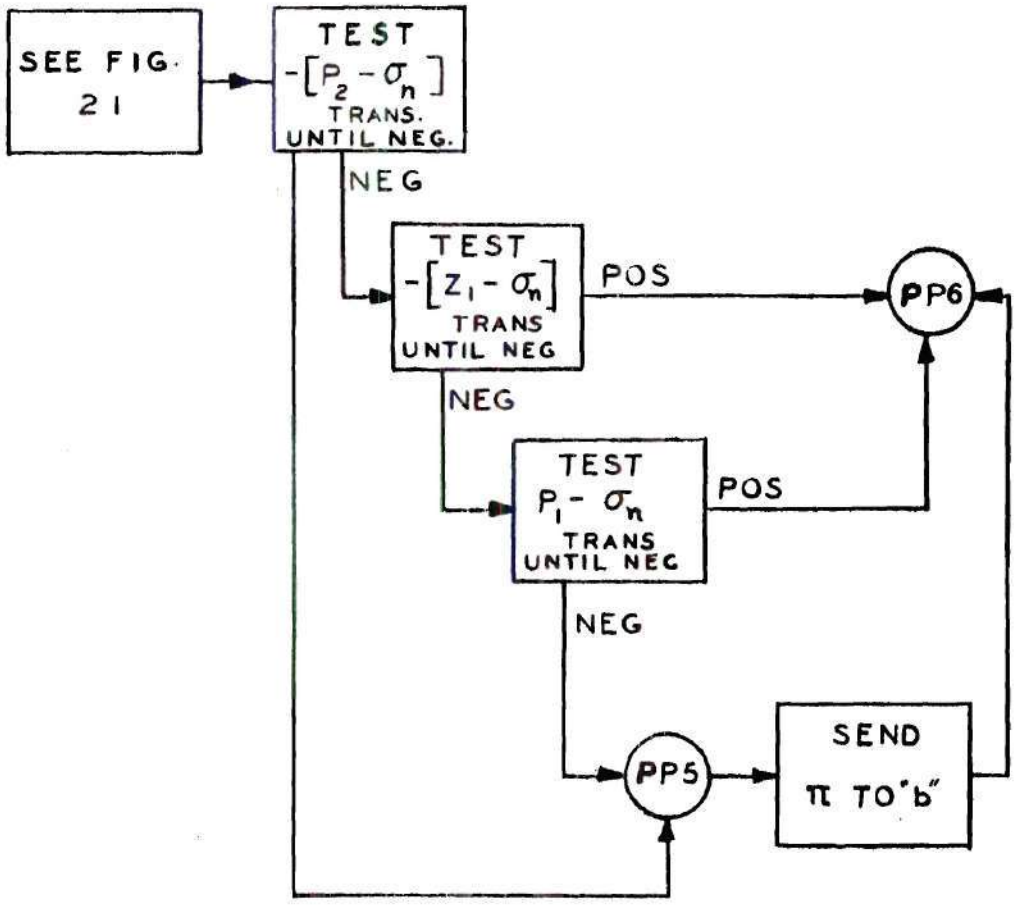

Figure 23. Flow Chart for Determining Region of Calculation.

compensating network which would have added a zero but no pole to the system.

Parallel "T" Compensated System. --The desired network was one giving derivative control, or one containing a zero, and a pole which was negligible in its effect on the system. The pole-zero configuration desired was as shown in Figure 26, page 56. The rules for approximating the root-locus can be applied to this pole-zero configuration giving the sketch shown in Figure 27, page 56. The root-locus for three poles and one zero was solved on the computer, where the open-loop system transfer function was

$$\frac{\theta_o(s)}{\theta_i(s)} = \frac{K(s + 18.0)(12.6)(23.2)}{s(s + 12.6)(s + 23.2)(18.0)} \tag{50}$$

The compensation network which was added to the system transfer function was not designed. References (9), (10) and (11) in the Bibliography give methods for the design of compensating networks for a-c systems.

The point of breakaway can be found in a manner similar to that applied to the previous problem. The equation to be solved is

$$\frac{1}{\sigma_1} + \frac{1}{\sigma_4} = \frac{1}{\sigma_2} + \frac{1}{\sigma_3} \tag{51}$$

where the σ's are as shown in Figure 28, page 58. The solution was

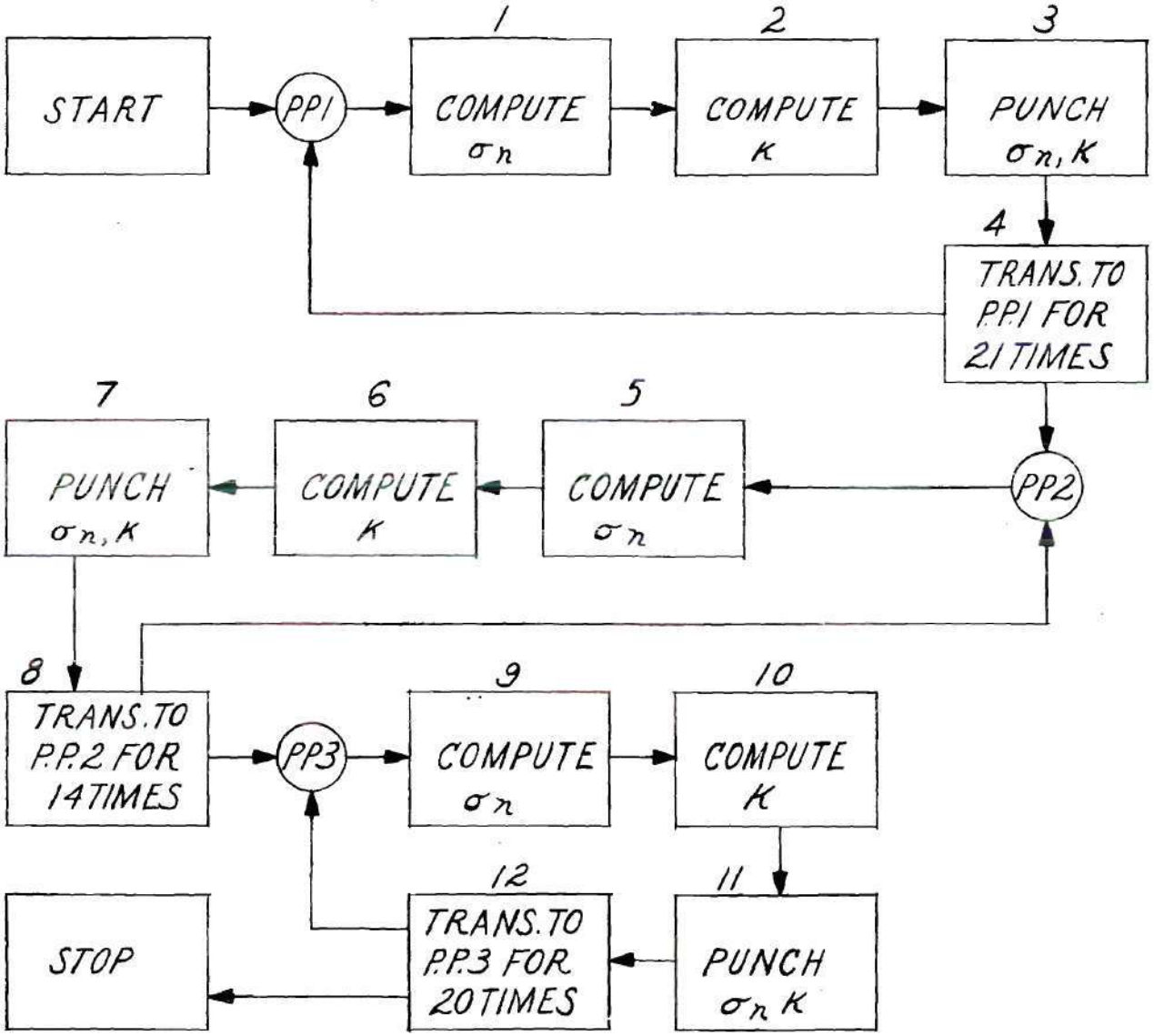

Figure 24. Flow Chart for Gain Solution Along the Real Axis for Four Poles - One Zero.

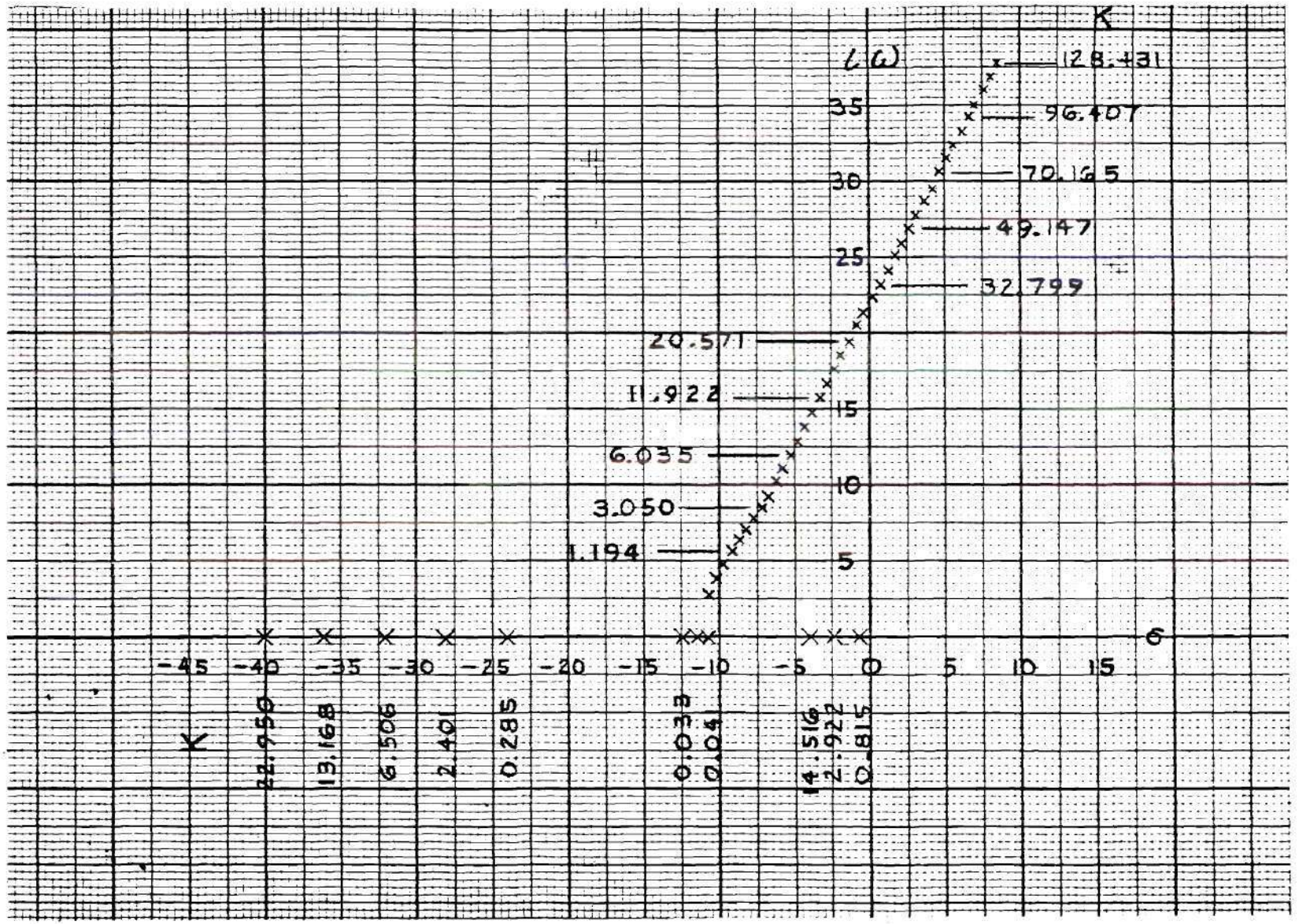

Figure 25. Compensated System Root-Locus Plot.

carried out in the same manner to an error less than 1×10^{-5}. The flow diagram for this solution is shown in Figure 29, page 59. The lower part of Table 11, page 81 in the Appendix, shows that three iterations were necessary to reduce the error below the limiting value.

The part of the root-locus containing imaginary parts lies in the region between p_1 and p_2. Hence the difficulties encountered in the previous problem are not met with here. The equation which must be satisfied is

$$\theta_1 - \phi_1 - \phi_2 - \phi_3 = \pi + 2n\pi \tag{52}$$

where "n" is equal to any integer or zero and the angles θ_1, ϕ_1, ϕ_2 and ϕ_3 are as indicated in Figure 28, page 58. The angles which the machines computes are ϕ_3, ϕ_2, θ_1 and $\pi - \phi_1$. Hence the equation which the machine solves is

$$\theta_1 + (180^\circ - \phi_1) - \phi_2 - \phi_3 - \pi = -\pi \tag{53}$$

Whereas in the two previous problems approximation occurred in the vertical plane for successive values in the horizontal plane, in this problem it seemed best to reverse this procedure due to the shape of the root-locus and approximate in the horizontal plane for successive values in the vertical plane. With these changes the program given on

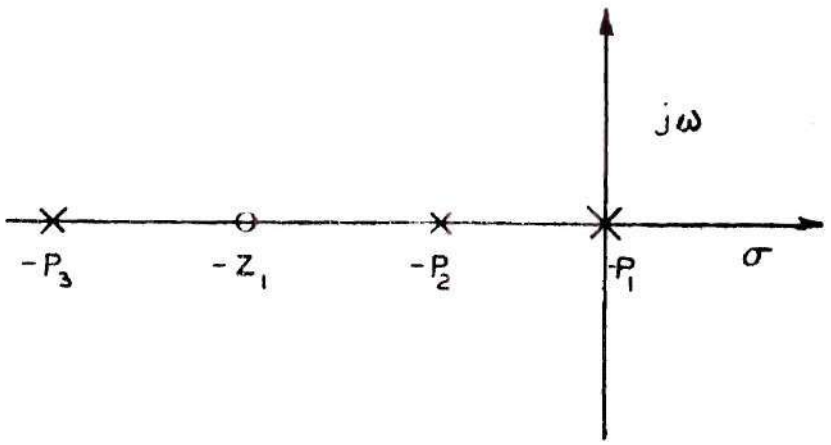

Figure 26. Pole-Zero Configuration with Proposed Compensation.

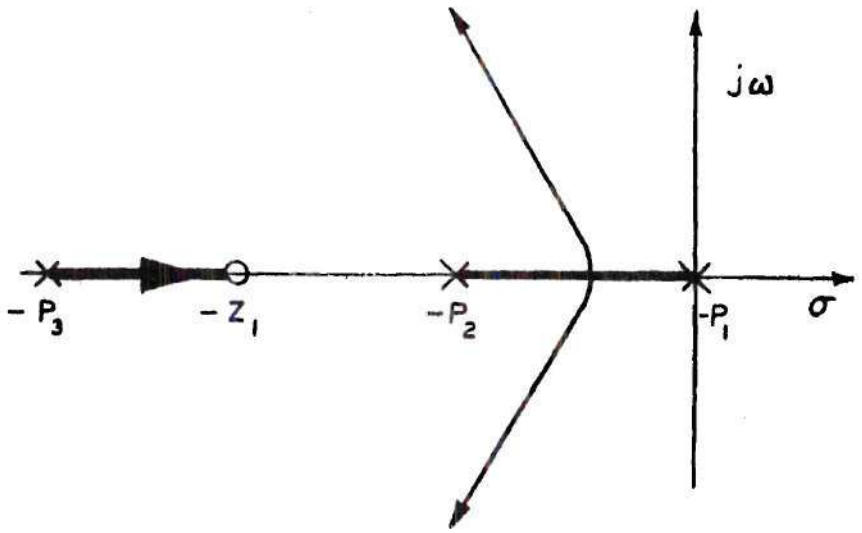

Figure 27. Sketch of Root-Locus for Three Poles - One Zero.

page 94 in the Appendix, is essentially the same as that for the uncompensated system. The flow chart is shown in Figure 30, page 60. The values computed are given in Table 9, pages 78 and 79 in the Appendix. Table 10, page 80 in the Appendix, gives the convergence for the computations and shows that two iterations were sufficient in most of the cases. Values of gain along the axis were not computed for this case. The same type of program used in the previous problem with slight changes would do for this solution.

The root-locus is plotted in Figure 31, page 61, with values of gain marked for every fourth point calculated. From the root-locus plot, it is evident that the system is stable for all values of gain with this type of compensation. Since a pole must appear for most physically realizable networks, the plot is only of theoretical interest in showing the asymptote which the root-locus tends to approach as the pole of the compensating network is pushed further to the left.

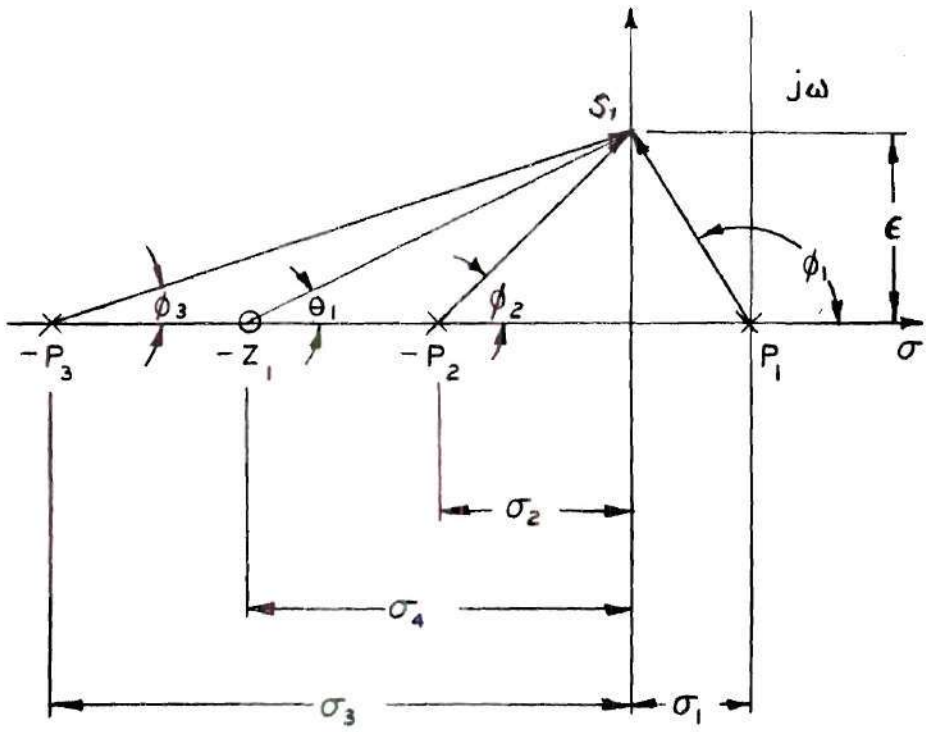

Figure 28. Determination of Breakaway Point.

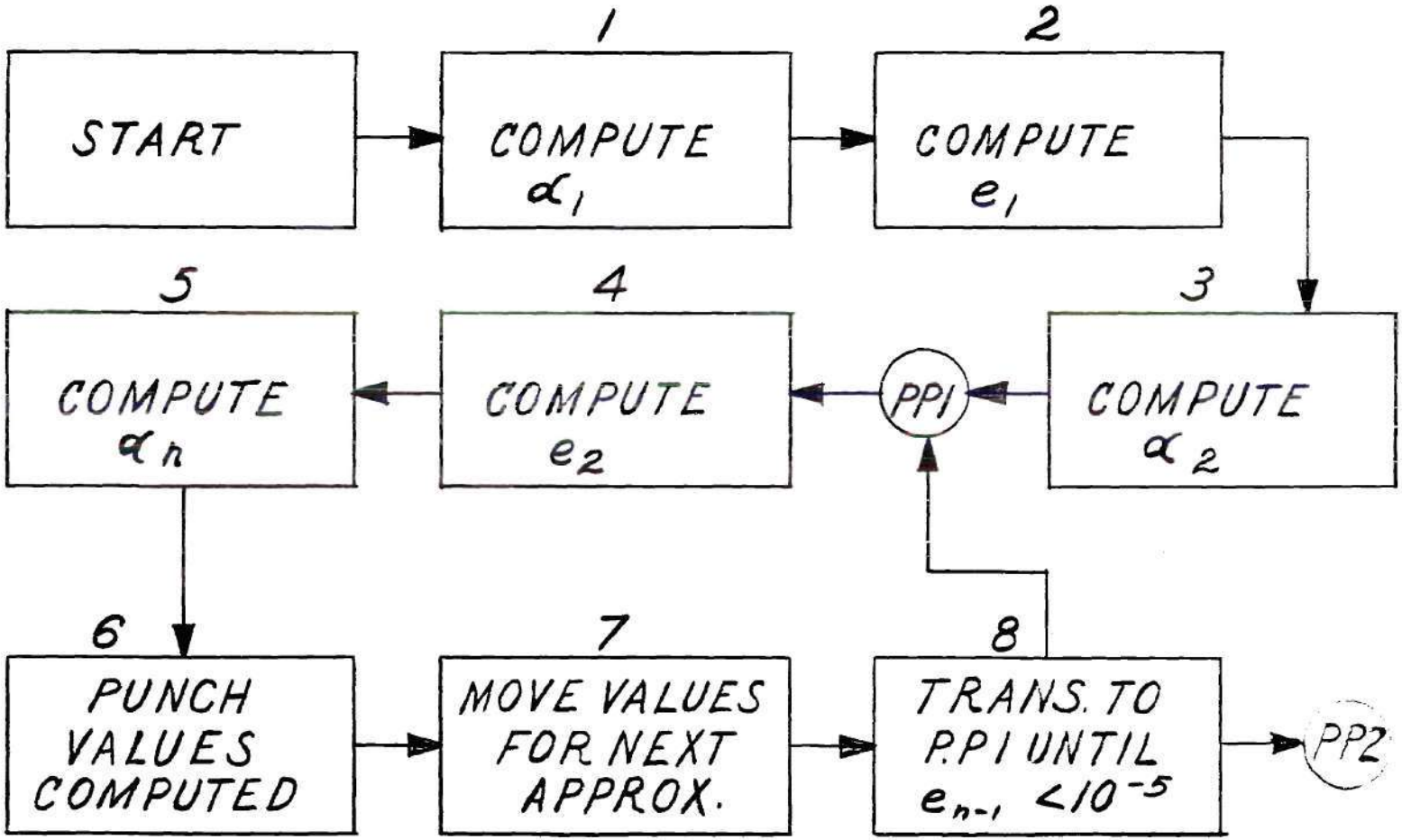

Figure 29. Flow Chart for Solution of Breakaway Point for Three Poles and One Zero.

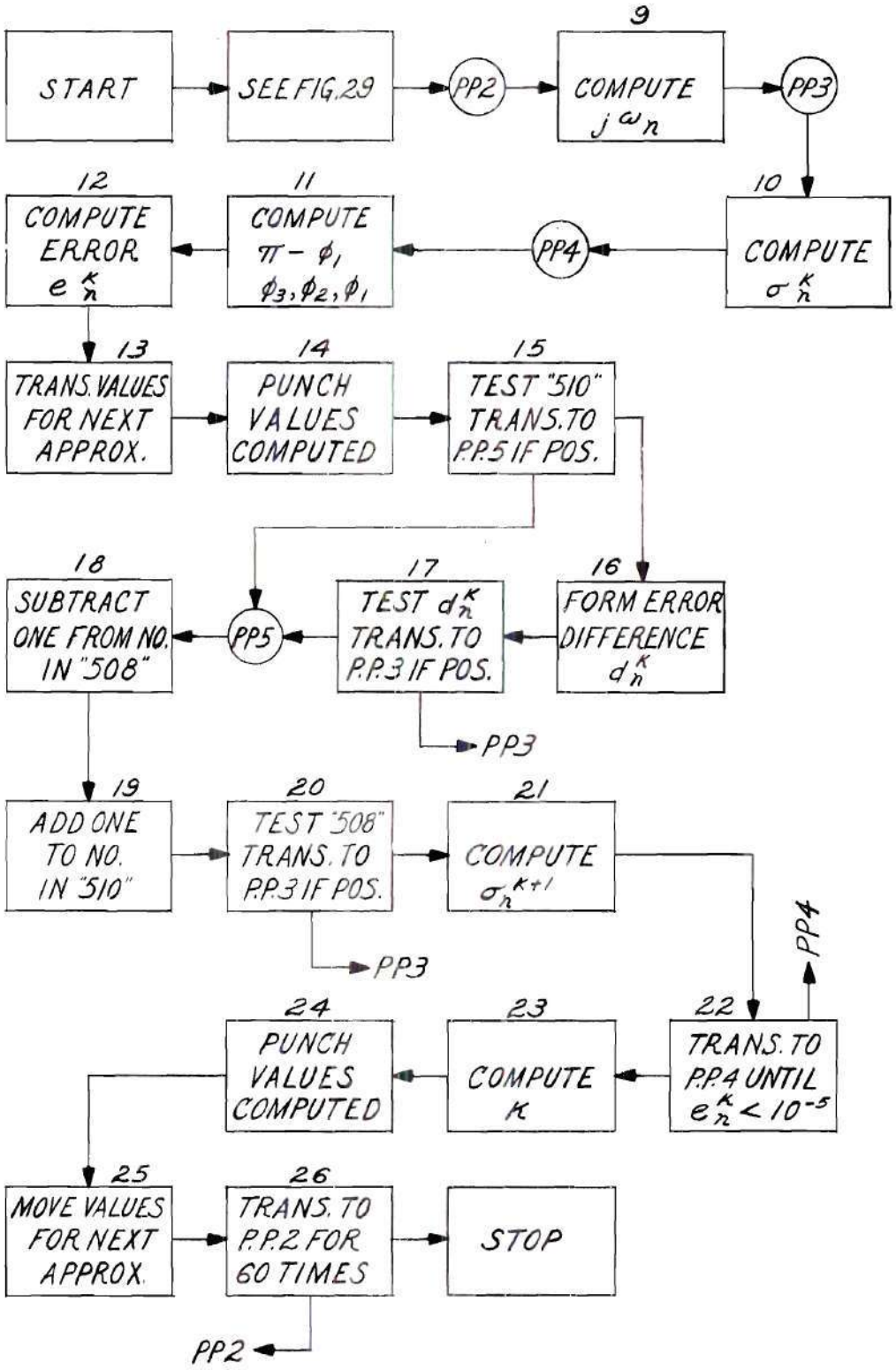

Figure 30. Flow Chart for Complex Solution of Root-Locus for Three Poles - One Zero.

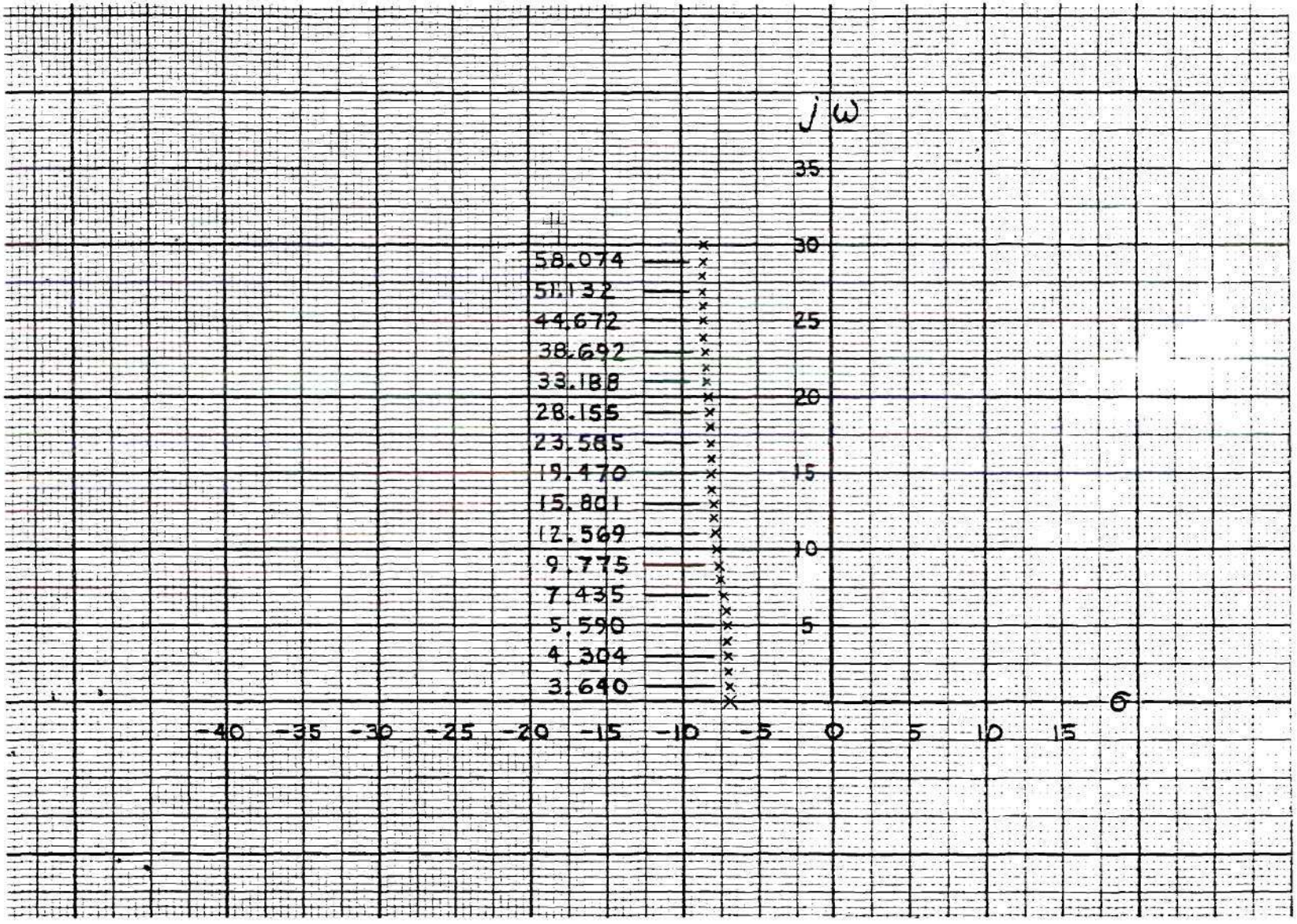

Figure 31. Parallel-T Compensated System Root-Locus Plot.

CHAPTER IV

CONCLUSIONS

The root-locus solution for three specific cases was obtained using the digital computer.

The number of decimal digits retained in computations was not necessary from a physical standpoint. The error limit established was desired as a check on the convergence of the approximating method. The convergence in all cases was rapid and direct, and would tend to be moreso as the error limit is reduced.

The computing time for any one problem was less than 15 minutes. However, programming took hours and for most practical situations is a major handicap of the method. If only one problem needed to be solved, it would hardly be worthwhile. If several problems of the same type were to be solved, then a digital computer might be favored.

The programs obtained were flexible as to the number of points computed, the interval between computations and the error limit. Also, changes in the poles and zeros could be made so long as the general properties of the root-locus were not altered. Thus, a definite advantage could be realized if the same problem had to be repeated many times with slight variations in the critical frequencies.

CHAPTER V

RECOMMENDATIONS

It is recommended that further study and investigation toward a more general solution be undertaken.

It is possible that a subroutine might be developed which would solve a large variety of pole-zero configurations by the insertion of only a minimum of necessary constants. Possibly two or three such subroutines would cover most practical situations, and thus eliminate all tedious programming operations.

APPENDIX

DERIVATION OF POINT OF BREAKAWAY

The point of breakaway will be derived for a system with three real open-loop poles. The graphical picture of the root-locus for such a system is given in Figures 11 and 12, page 29. The poles of the system are shown as p_1, p_2 and p_3. The point of breakaway is indicated as $-\alpha$ in Figure 11. Assume that the point "s_1" is chosen a small distance, ϵ_1 off the axis from the point $-\alpha$. The transition from $-\alpha$ to "s_1" must result in a zero net change in the angle of F(s). The sum of the changes in the angles from the various open-loop critical frequencies is equated to zero, with the appropriate signs depending on the direction of the change and whether the critical frequency is a pole or zero. In this example, Figure 12, page 29, indicates that

$$\phi_3 + \phi_2 - \theta = 0 \tag{54}$$

The small angles can be replaced by the corresponding tangents as

$$\frac{\epsilon}{p_3 - \alpha} + \frac{\epsilon}{p_2 - \alpha} - \frac{\epsilon}{\alpha} = 0 \tag{55}$$

After cancellation of ϵ, the equation can be reduced as follows:

$$\frac{1}{p_3 - \alpha} + \frac{1}{p_2 - \alpha} = \frac{1}{\alpha} \tag{56}$$

$$3\alpha^2 - 2\alpha p_2 - 2\alpha p_3 + p_2 p_3 = 0 \tag{57}$$

From which

$$\alpha = 1/3 (p_2 + p_3) \pm \sqrt{1/9 (p_2 + p_3)^2 - 1/3 p_2 p_3} \tag{58}$$

is the required equation to be solved.

A quick approximation with the following poles

$$p_1 = 0.0 \tag{59}$$

$$p_2 = 12.6 \tag{60}$$

$$p_3 = 23.2 \tag{61}$$

shows that

$$\alpha \cong 12 \pm 6.8 = 18.8; 5.2 \tag{62}$$

Since the one answer, 18.8, lies on a portion of the real axis, which contains no loci, only the negative sign is retained for the computation. Hence, the point of breakaway is determined by

$$\alpha = 1/3 (p_2 + p_3) - \sqrt{1/9 (p_2 + p_3)^2 - 1/3 p_2 p_3} \tag{63}$$

Table 1. Uncompensated System Measurements

f(cps)	ω(rad/sec)	A_{in}	A_{out}	K	20 log K
0.15	0.945	0.36	37.0	102.0	40.2
0.20	1.258	0.38	33.5	88.2	38.9
0.30	1.888	0.38	26.0	68.4	36.7
0.40	2.515	0.40	23.2	58.0	35.3
0.50	3.145	0.40	20.5	51.2	34.1
0.60	3.755	0.40	18.0	45.0	33.1
0.70	4.40	0.39	15.8	40.5	32.1
0.80	5.03	0.41	15.0	37.1	31.3
0.90	5.66	1.40	39.0	27.9	28.9
1.00	6.28	1.45	37.0	25.5	28.1
1.10	6.93	1.50	34.5	23.0	27.2
1.20	7.56	1.90	39.0	20.5	26.2
1.30	8.18	1.90	37.5	19.8	25.9
1.40	8.82	1.90	35.0	18.4	25.3
1.50	9.44	1.98	32.0	17.0	24.6
1.60	10.08	1.85	29.0	15.7	23.9
1.70	10.70	2.15	31.0	14.4	23.1
1.80	11.33	2.10	28.5	13.6	22.7
1.90	11.97	2.05	25.0	12.2	21.7
2.00	12.59	2.00	22.0	11.0	20.8
2.5	15.73	1.70	12.5	7.4	17.3
3.0	18.88	1.30	7.0	5.4	14.6
3.5	22.00	1.25	4.5	3.6	11.1
4.0	25.15	1.15	2.60	2.26	7.08
4.5	28.30	1.00	1.80	1.80	5.10
5.0	31.45	1.00	1.30	1.30	2.26
5.5	34.60	1.00	1.00	1.00	0.00
6.0	37.55	0.95	0.80	0.84	-1.50
7.0	44.0	0.95	0.50	0.53	-5.58
8.0	50.3	0.95	0.30	0.32	-10.00
9.0	56.6	0.95	0.25	0.26	-11.60
10.0	62.8	0.95	0.20	0.21	-13.60

Table 2. Compensated System Measurements

f(cps)	ω(rad/sec)	A_{out}	A_{in}	K	20 log K
0.15	0.945	30.0	0.34	88.3	38.9
0.20	1.258	24.0	0.34	70.6	36.9
0.30	1.888	35.0	0.80	43.7	32.8
0.40	2.515	37.0	1.00	37.0	31.3
0.50	3.145	36.0	1.15	31.3	29.9
0.60	3.755	35.0	1.25	28.0	28.9
0.70	4.40	34.5	1.35	25.5	28.1
0.80	5.03	33.5	1.40	23.9	27.5
0.90	5.66	32.5	1.45	22.4	27.0
1.00	6.28	30.5	1.53	19.9	26.0
1.10	6.93	29.0	1.55	18.7	25.4
1.20	7.56	28.0	1.60	17.5	24.9
1.30	8.18	26.0	1.60	16.3	24.2
1.40	8.82	23.0	1.60	14.4	23.1
1.50	9.44	21.0	1.50	14.0	22.9
1.60	10.08	20.0	1.50	13.3	22.5
1.70	10.70	17.5	1.60	10.9	20.7
1.80	11.33	16.0	1.50	10.7	20.6
1.90	11.97	14.0	1.40	10.0	20.0
2.00	12.59	13.0	1.40	9.28	19.4
2.5	15.73	16.0	2.10	7.62	17.7
3.0	18.88	10.0	2.00	5.00	14.0
3.5	22.00	12.0	3.00	4.00	12.1
4.0	25.15	8.0	2.90	2.76	8.81
4.5	28.30	7.0	2.90	2.41	7.64
5.0	31.45	5.0	2.80	1.78	5.00
5.5	34.60	3.40	2.80	1.21	1.40
6.0	37.55	2.70	2.80	0.964	-0.32
6.5	40.8	2.10	2.70	0.777	-2.20
7.0	44.0	1.80	2.70	0.667	-7.06
8.0	50.3	1.20	2.70	0.444	-7.82
9.0	56.6	1.10	2.70	0.407	-10.60
10.0	62.8	0.80	2.70	0.296	-10.60
13.0	81.8	0.30	2.70	0.111	-19.10

Table 3. Calculated Complex Values for Three Pole Root-Locus

σ_n	$j\omega_n^m$	K	$e_n^{m-1} \times 10^{-5}$
-5.228	0.000	-------	------
-4.728	4.568	3.895	0.05
-4.228	6.575	5.716	-0.04
-3.728	8.191	7.854	<0.01
-3.228	9.616	10.328	0.05
-2.728	10.924	13.160	<0.01
-2.228	12.153	16.370	<0.01
-1.728	13.325	19.978	<0.01
-1.228	14.455	24.005	<0.01
-0.728	15.550	28.471	<0.01
-0.228	16.618	33.398	<0.01
0.272	17.664	38.805	0.01
0.772	18.692	44.713	0.01
1.272	19.704	51.142	0.01
1.772	20.703	58.114	<0.01
2.272	21.691	65.648	<0.01
2.772	22.669	73.766	<0.01
3.272	23.638	82.487	<0.01
3.772	24.599	91.833	<0.01
4.272	25.553	101.823	<0.01
4.772	26.502	112.478	<0.01
5.272	27.444	123.819	0.01
5.772	28.383	135.867	<0.01
6.272	29.316	148.641	<0.01
6.772	30.246	162.163	-0.01
7.272	31.172	176.452	0.01
7.772	32.094	191.530	<0.01
8.272	33.014	207.417	0.93
8.772	33.930	224.133	0.86
9.272	34.844	241.699	0.78
9.772	35.756	260.136	0.73

Table 4. Calculated Real Values for Three Pole Root-Locus

$-\sigma$	K	$-\sigma$	K
0.50	0.470	12.00	0.276
1.00	0.881	12.50	0.046
1.50	1.236	24.20	0.960
2.00	1.537	25.20	2.172
2.50	1.788	26.20	3.657
3.00	1.990	27.20	5.434
3.50	2.146	28.20	7.525
4.00	2.259	29.20	9.949
4.50	2.332	30.20	12.728
5.00	2.366	31.20	15.882
5.50	2.364	32.20	19.431
6.00	2.330	33.20	23.396
6.50	2.265	34.20	27.798
7.00	2.172	35.20	32.657
7.50	2.054	36.20	37.993
8.00	1.914	37.20	43.828
8.50	1.753	38.20	50.181
9.00	1.574	39.20	57.073
9.50	1.380	40.20	64.525
10.00	1.174	41.20	72.557
10.50	0.958	42.20	81.189
11.00	0.735	43.20	90.443
11.50	0.506		

Table 5. Rapidity of Convergence for Three Pole Root-Locus Solution

σ_n	$j\omega_n^k$	$e_n^k \times 10^{-5}$
-4.728	0.200	605.1
	0.400	1198.7
	0.600	1769.5
	0.800	2306.8
	1.000	2800.9
	1.200	3242.8
	1.400	3625.0
	1.600	3941.0
	1.800	4185.9
	2.000	4355.7
	2.200	4448.2
	2.400	4462.0
	2.600	4396.9
	2.800	4253.6
	3.000	4033.9
	3.200	3739.8
	5.744	-4967.3
	4.293	964.9
	4.529	143.7
	4.570	- 6.80
	4.568	0.05
-4.228	4.768	8147.0
	4.568	8831.3
	7.149	-3037.2
	6.489	446.8
	6.573	10.32
	6.575	- 0.04

Table 5. Continued

Rapidity of Convergence for Three Pole Root-Locus Solution

σ_n	$j\omega_n^k$	$e_n^k \times 10^{-5}$
-2.228	11.124	6510.5
	10.924	7810.9
	12.125	172.2
	12.152	4.90
	12.153	0.01
0.772	17.864	4533.2
	17.664	5661.6
	18.668	129.9
	18.691	3.82
	18.692	0.01

Table 6. Calculated Complex Values for Four Pole-One Zero Root-Locus

σ_n	$j\omega_n^m$	K	$e_n^{m-1} \times 10^{-5}$
-11.364	0.000	-------	-----
-10.864	2.709	0.282	-0.01
-10.364	3.868	0.548	0.02
- 9.864	4.790	0.852	0.01
- 9.364	5.605	1.194	0.74
- 8.864	6.369	1.577	0.35
- 8.364	7.115	2.007	0.20
- 7.864	7.866	2.494	0.11
- 7.364	8.637	3.050	0.80
- 6.864	9.439	3.692	0.04
- 6.364	10.273	4.437	0.02
- 5.864	11.140	5.303	0.01
- 5.364	12.033	6.305	0.01
- 4.864	12.946	7.457	0.02
- 4.364	13.874	8.770	0.01
- 3.864	14.811	10.255	0.01
- 3.364	15.752	11.922	0.01
- 2.864	16.696	13.778	0.01
- 2.364	17.640	15.833	0.02
- 1.864	18.582	18.095	0.01
- 1.364	19.522	20.571	0.01
- 0.864	20.459	23.272	0.02
- 0.364	21.393	26.204	0.01
0.136	22.324	29.377	0.01
0.636	23.252	32.799	0.01
1.136	24.177	36.479	-0.01

Table 6. Continued

Calculated Complex Values for Four Pole-One Zero Root-Locus

σ_n	$j\omega_n^m$	K	$e_n^{m-1} \times 10^{-5}$
1.636	25.100	40.424	-0.02
2.136	26.019	44.644	-0.01
2.636	26.936	49.147	-0.01
3.136	27.850	53.942	-0.01
3.636	28.762	59.038	0.01
4.136	29.671	64.442	0.01
4.636	30.579	70.165	0.01
5.136	31.484	76.214	0.01
5.636	32.388	82.598	0.96
6.136	33.290	89.326	0.86
6.636	34.190	96.407	0.83
7.136	35.089	103.849	0.75
7.636	35.987	111.661	0.71
8.136	36.883	119.853	0.66
8.636	37.778	128.431	0.61

Table 7. Calculated Real Values for Four Pole-One Zero Root-Locus

$-\sigma$	K	$-\sigma$	K
0.200	0.200	11.700	0.052
0.400	0.402	11.900	0.047
0.600	0.607	12.100	0.041
0.800	0.815	12.300	0.033
1.000	1.028	12.500	0.023
1.200	1.248	12.700	0.012
1.400	1.478	12.900	0.000
1.600	1.721	24.200	0.285
1.800	1.981	25.200	0.659
2.000	2.262	26.200	1.130
2.200	2.572	27.200	1.708
2.400	2.922	28.200	2.401
2.600	3.324	29.200	3.219
2.800	3.801	30.200	4.170
3.000	4.386	31.200	5.262
3.200	5.134	32.200	6.506
3.400	6.143	33.200	7.910
3.600	7.606	34.200	9.482
3.800	9.968	35.200	11.232
4.000	14.516	36.200	13.168
4.200	27.225	37.200	15.299
10.300	0.017	38.200	17.634
10.500	0.031	39.200	20.181
10.700	0.041	40.200	22.950
10.900	0.048	41.200	25.949
11.100	0.053	42.200	29.188
11.300	0.055	43.200	32.674
11.500	0.054		

Table 8.

Rapidity of Convergence for Four Pole-One Zero Root-Locus Solution

σ_n	$j\omega_n^k$	$e_n^k \times 10^{-5}$
-10.864	0.200	12940.0
	0.400	23090.0
	0.600	29310.0
	0.800	31950.0
	1.000	31910.0
	1.200	30050.0
	1.400	27030.0
	1.600	23340.0
	2.862	- 3069.0
	2.715	- 123.0
	2.709	1.50
	2.709	- 0.01
-10.364	2.909	22220.0
	2.709	27540.0
	3.745	2565.0
	3.851	337.0
	3.868	4.97
	3.868	0.02
- 9.364	4.990	11650.0
	4.790	15850.0
	5.545	1056.0
	5.599	102.0
	5.605	0.74

Table 8. Continued

Rapidity of Convergence for Four Pole-One Zero Root-Locus Solution

σ_n	$j\omega_n^k$	$e_n^k \times 10^{-5}$
- 5.364	11.340	5714.0
	11.140	7433.0
	12.004	227.0
	12.032	9.29
	12.033	0.01
5.136	30.779	2734.0
	30.579	3525.0
	31.471	52.55
	31.484	1.02
	31.484	<0.01

Table 9.

Calculated Complex Values for Three Pole-One Zero Root-Locus

$-\sigma_n^m$	$j\omega_n$	K	$e_n^{m-1} \times 10^{-5}$
6.863	0.00	------	------
6.866	0.50	3.577	-0.004
6.875	1.00	3.640	0.307
6.889	1.50	3.745	0.663
6.910	2.00	3.891	-0.001
6.936	2.50	4.077	-0.004
6.967	3.00	4.304	0.005
7.003	3.50	4.569	0.003
7.053	4.00	4.873	0.01
7.086	4.50	5.214	-0.004
7.134	5.00	5.590	0.96
7.184	5.50	6.002	0.66
7.236	6.00	6.447	0.38
7.290	6.50	6.925	0.12
7.345	7.00	7.435	-0.12
7.401	7.50	7.975	-0.32
7.457	8.00	8.546	-0.47
7.513	8.50	9.146	-0.59
7.569	9.00	9.775	-0.66
7.623	9.50	10.432	-0.72
7.677	10.00	11.117	-0.74
7.729	10.50	11.829	-0.74
7.779	11.00	12.569	-0.71
7.828	11.50	13.336	-0.68
7.875	12.00	14.131	-0.65
7.920	12.50	14.952	-0.60
7.963	13.00	15.801	-0.56
8.004	13.50	16.677	-0.50
8.044	14.00	17.580	-0.45
8.081	14.50	18.511	-0.41

Table 9. Continued

Calculated Complex Values for Three Pole-One Zero Root-Locus

$-\sigma_n^m$	$j\omega_n$	K	$e_n^{m-1} \times 10^{-5}$
8.117	15.00	19.470	-0.38
8.151	15.50	20.457	-0.34
8.183	16.00	21.471	-0.30
8.213	16.50	22.514	-0.26
8.242	17.00	23.585	-0.25
8.270	17.50	24.684	-0.20
8.296	18.00	25.813	-0.19
8.320	18.50	26.969	-0.16
8.344	19.00	28.155	-0.15
8.366	19.50	29.370	-0.13
8.387	20.00	30.613	-0.12
8.407	20.50	31.886	-0.08
8.426	21.00	33.188	-0.07
8.443	21.50	34.520	-0.08
8.460	22.00	35.881	-0.07
8.477	22.50	37.272	-0.06
8.492	23.00	38.692	-0.06
8.507	23.50	40.143	-0.04
8.520	24.00	41.623	-0.05
8.534	24.50	43.132	-0.04
8.546	25.00	44.672	-0.02
8.558	25.50	46.242	-0.02
8.570	26.00	47.842	-0.04
8.580	26.50	49.472	-0.03
8.591	27.00	51.132	-0.02
8.601	27.50	52.822	-0.03
8.610	28.00	54.542	-0.02
8.619	28.50	56.293	-0.01
8.628	29.00	58.074	-0.01
8.636	29.50	59.885	-0.01
8.644	30.00	61.726	-0.01

Table 10.

Rapidity of Convergence for Three Pole-One Zero Root-Locus Solution

$j\omega_n$	$-\sigma_n^k$	$e_n^k \times 10^{-5}$
0.500	7.06262	- 466.50
	7.26262	- 949.90
	7.46262	-1447.0
	7.66262	-1960.0
	6.89949	- 79.66
	6.86717	- 3.64
	6.86562	- 0.004
1.000	7.06562	- 886.3
	6.86562	41.3
	6.87452	0.31
1.500	7.07459	-1246.4
	6.87464	98.8
	6.88932	0.66
5.000	7.28650	-2138.2
	7.08650	658.9
	7.13361	0.96
10.000	7.82318	-1820.3
	7.62310	665.5
	7.67666	- 0.74
20.000	8.56579	-1468.2
	8.36577	172.4
	8.38679	- 0.12

Table 11. Rapidity of Convergence to Breakaway Point

Solution for Four Poles-One Zero

α_K	$e_n \times 10^{-5}$
12.000000	-13976.00
11.380123	-72267.00
11.365028	- 1718.00
11.363610	- 147.00
11.363615	0.47

Solution for Three Poles-One Zero

α_K	$e_n \times 10^{-5}$
6.8000000	2629.80
6.8632972	295.50
6.8626134	- 3.23
6.8626153	0.008

Table 12.

Program for Root-Locus Solution of Uncompensated System

Box No.	Card No.	Order	Remarks
	01	0199 in columns, 77, 78, 79, and 80 Problem No.	
1	02	1 500 501 550	Compute α
1	03	3 550 503 551	
1	04	2 551 551 552	
1	05	2 500 501 553	
1	06	-3 553 503 554	
1	07	1 552 554 555	
1	08	0 300 555 555	
1	09	-1 551 555 556	
2	10	7 300 556 556	Punch α
	11	9 800 002 000	Program Point 2
3	12	1 504 556 580	Compute σ_n
4	13	1 500 580 557	$P_3 - \sigma_n$
4	14	1 501 580 558	$P_2 - \sigma_n$
	15	9 800 003 000	Program Point 3
5	16	1 561 510 562	Compute $j\omega_n^K$
	17	9 800 004 000	Program Point 4
6	18	3 562 557 575	Compute ϕ_3
6	19	3 562 558 576	ϕ_2
6	20	3 562 580 577	
6	21	0 305 575 585	
6	22	0 305 576 586	
6	23	0 305 577 587	θ OR ϕ_1
7	24	1 900 900 508	Send Zero to "508"
8	25	8 700 580 005	Test σ_n
9	26	1 900 506 508	Send π to "508"
	27	9 800 005 000	Program Point 5
10	28	1 585 586 566	Compute a_n^K
10	29	1 566 587 566	
10	30	1 566 508 566	

Table 12. Continued

Program for Root-Locus Solution of Uncompensated System

Box No.	Card No.	Order			Remarks
11	31	1	506	566 565	Compute e_n^K
12	32	7	201	561 560	Transfer $j\omega_n^K$ to $j\omega_n^{K-1}$
12	33	7	201	562 561	$j\omega_n^{K+1}$ to $j\omega_n^K$
12	34	7	201	564 563	e_n^K to e_n^{K-1}
12	35	7	201	565 564	e_n^{K+1} to e_n^K
13	36	7	300	560 564	Punch $j\omega_n^K$, $j\omega_n^{K+1}$, e_n^K, e_n^{K+1}
14	37	8	700	512 006	Test "512" to P. P. 6 if pos.
15	38	-1	564	563 509	Compute d_n^{K+1}
16	39	8	700	509 003	Test "509" to P. P. 3 if pos.
	40	9	800	006 000	Program Point 6
17	41	-1	511	901 511	Subtract one from "511"
18	42	1	512	901 512	Add one to "512"
19	43	8	700	511 003	Test "511" to P. P. 3 if pos.
20	44	-2	560	564 567	Compute $j\omega_n^{K+2}$
20	45	4	561	563 567	
20	46	-1	563	564 568	
20	47	3	567	568 562	
21	48	8	845	564 004	Trans. to P. P. 4 until $e_n^K < 10^{-5}$
22	49	2	580	580 569	Compute K
22	50	2	557	557 570	
22	51	2	558	558 571	
22	52	2	562	562 572	
22	53	1	572	569 569	
22	54	1	572	570 570	
22	55	1	572	571 571	
22	56	0	300	569 569	
22	57	0	300	570 570	
22	58	0	300	571 571	
22	59	2	569	570 573	
22	60	2	573	571 573	

Table 12. Continued

Program for Root-Locus Solution of Uncompensated System

Box No.	Card No.	Order			Remark
22	61	3	573	553 582	
23	62	7	201	562 581	Transfer $j\omega_n^m$ to punch
23	63	7	201	564 583	e_n^{m-1}
23	64	7	300	580 583	Punch σ_n, $j\omega_n^m$, K, e_n^{m-1}
24	65	-2	901	580 556	Transfer $-\sigma_n$ to σ_{n-1}
24	66	7	201	562 561	$j\omega_n^m$ to $j\omega_{n-1}^m$,
25	67	8	100	030 002	Transfer to P. P. 2 thirty times
26	68	0	000	000 000	Stop
	69	2	320	000 051	500:p_3
		1	260	000 051	501:p_2
		0	000	000 000	502:p_1
		3	000	000 050	503:3.0
		5	000	000 049	504:0.5
	70	3	141	592 750	506: π
		-2	000	000 050	507:-2.0
		0	000	000 000	508:0.0
		1	000	000 050	509:1.0
	71	2	000	000 049	510:0.2
		3	000	000 050	511:3.0
		-2	000	000 050	512:-2.0
	72	0	000	000 000	501:0.0
	73	0	000	000 000	504:0.0
	74		000		Zero in col. 7, 8, 9, and 10

Table 13.

Program for Gain Solution of Uncompensated System along Real Axis

Box No.	Card No.	Order	Remarks
	01	0199 in columns 77, 78, 79 and 80	Problem No.
	02	9 800 001 000	Program Point 1
1	03	1 500 504 520	Compute σ_n
2	04	-1 501 520 521	$p_3 - \sigma_n$
2	05	-1 502 520 522	$p_2 - \sigma_n$
3	07	2 520 521 519	Compute K
3	08	2 519 522 519	
3	09	3 519 501 519	
3	10	3 519 502 519	
4	11	7 300 519 520	Punch σ_n, K
5	12	7 201 520 500	Trans. σ_n to σ_n -1
6	13	8 100 025 001	Trans. to P. P. 1 25 times
	14	9 800 002 000	Program Point 2
7	15	1 505 901 520	Compute: σ_n
8	16	-1 520 501 521	$\sigma_n - p_3$
8	17	-1 520 502 522	$\sigma_n - p_2$
9	18	2 520 521 519	Compute K
9	19	2 519 522 519	
9	20	3 519 501 519	
9	21	3 519 502 519	
10	22	7 300 519 520	Punch σ_n, K
11	23	7 201 520 505	Trans. σ_n to σ_n -1
12	24	8 200 020 002	Trans. to P. P. 2 20 times
13	25	0 000 000 000	Stop
	26	0 000 000 000	500: 0.0
		2 320 000 051	501: p_3
		1 260 000 051	502: p_2
		0 000 000 000	503: p_1
		5 000 000 049	504: 0.5
		2 320 000 051	505: p_3
	27	0000 in columns 7, 8, 9 and 10	

Table 14.

Program for Root-Locus Solution of Compensated System

Box No.	Card No.	Order	Remarks
	01	0199 in columns 77, 78, 79 and 80	Problem No.
1	02	1 502 503 520	Compute α_1
1	03	3 520 505 520	
2	04	-1 520 500 521	Compute e_1
2	05	-1 520 502 522	
2	06	-1 503 520 523	
2	07	-1 504 520 524	
2	08	3 901 520 525	
2	09	3 901 521 526	
2	10	3 201 522 527	
2	11	3 901 523 528	
2	12	3 901 524 529	
2	13	1 525 527 530	
2	14	-1 530 526 530	
2	15	-1 530 528 530	
2	16	-1 530 529 530	
3	17	1 520 506 532	Compute α_2
	18	9 800 001 000	Program Point 1
4	19	-1 532 500 521	Compute e_2
4	20	-1 532 502 522	
4	21	-1 503 532 523	
4	22	-1 504 532 524	
4	23	3 901 532 525	
4	24	3 901 521 526	
4	25	3 901 522 527	
4	26	3 901 523 528	
4	27	3 901 524 529	
4	28	1 525 527 531	

Table 14. Continued

Program for Root-Locus Solution of Compensated System

Box No.	Card No.	Order			Remarks
4	29	-1	531	526 531	
4	30	-1	531	528 531	
4	31	-1	531	529 531	
5	32	-2	520	531 534	Compute α_{K+1}
5	33	4	530	532 534	
5	34	-1	530	531 535	
5	35	3	534	535 533	
6	36	7	300	530 533	Punch $e_{K-1}, e_K, \alpha_K, \alpha_{K+1}$
7	37	7	201	531 530	Transfer e_K to e_{K-1}
7	38	7	201	532 520	α_K to α_{K-1}
7	39	7	201	533 532	α_{K+1} to α_K
8	40	8	845	530 001	Trans. to P. P. 1 until $e_K < 10^{-5}$
	41	9	800	002 000	Program Point 2
10	42	-1	506	532 580	Compute σ_n
11	43	1	504	580 537	$p_4 - \sigma_n$
11	44	1	503	580 538	$p_3 - \sigma_n$
11	45	1	502	580 539	$p_2 - \sigma_n$
11	46	1	500	580 540	$z_1 - \sigma_n$
	47	9	800	003 000	Program Point 3
12	48	1	561	507 562	Compute $j\omega_n^K$
	49	9	800	004 000	Program Point 4
13	50	3	562	537 541	Compute
13	51	3	562	538 542	ϕ_4
13	52	3	562	539 543	ϕ_3
13	53	3	562	540 544	ϕ_2 OR $\pi - \phi_2$
13	54	3	562	580 545	θ_1 OR $\pi - \theta_1$
13	55	0	305	541 546	ϕ_1 OR $\pi - \phi_1$
13	56	0	305	542 547	
13	57	0	305	543 548	
13	58	0	305	544 549	

Table 14. Continued

Program for Root-Locus Solution of Compensated System

Box No.	Card No.		Order			Remarks
13	59	0	305	545	550	
14	60	1	900	900	551	Send Zero to Address "b"
15	61	-1	901	539	552	Determination
15	62	8	700	552	005	of
15	63	-2	901	540	553	Region
15	64	8	700	553	006	of
15	65	8	700	580	006	Computation
15	66	9	800	005	000	
15	67	1	900	508	551	
	68	9	800	006	000	Program Point 6
16	69	-1	549	546	566	Compute q_n^k
16	70	-1	566	547	566	
16	71	-1	566	548	566	
16	72	-1	566	550	566	
16	73	-1	566	551	566	
17	74	1	508	566	565	Compute e_n^k
18	75	7	201	561	560	Transfer $j\omega_n^{k-1}$ to $j\omega_n^{k-2}$
18	76	7	201	562	561	$j\omega_n^k$ to $j\omega_n^{k-1}$
18	77	7	201	564	563	e_n^{k-1} to e_n^{k-2}
18	78	7	201	565	564	e_n^k to e_n^{k-1}
18	79	7	300	560	564	Punch $j\omega_n^{k-1}$, $j\omega_n^k$, e_n^{k-1}, e_n^k
19	80	8	700	509	007	Test "509" to P. P. 7 if pos.
20	81	-1	564	563	510	Compute d_n^k
21	82	8	700	510	003	Test d_n^k to P. P. 3 if pos.
	83	9	800	007	000	Program Point 7
22	84	-1	511	901	511	Subtract one from "511"
23	85	1	509	901	509	Add one to "509"
24	86	8	700	511	003	Test "511" to P. P. 3 if pos.
25	87	-2	560	564	567	Compute $j\omega_n^{k+1}$
25	88	4	561	563	567	
25	89	-1	563	564	568	
25	90	3	567	568	562	

Table 14. Continued

Program for Root-Locus Solution of Compensated System

Box No.	Card No.	Order	Remarks
26	91	8 845 564 004	Trans. to P. P. 4 until $e_n^K < 10^{-5}$
27	92	2 580 580 554	Compute K
27	93	2 537 537 555	
27	94	2 538 538 556	
27	95	2 539 539 557	
27	96	2 540 540 558	
27	97	2 562 562 559	
27	98	1 559 554 554	
27	99	1 559 555 555	
27	100	1 559 556 556	
27	101	1 559 557 557	
27	102	1 559 558 558	
27	103	0 300 554 554	
27	104	0 300 555 555	
27	105	0 300 556 556	
27	106	0 300 557 557	
27	107	0 300 558 558	
27	108	2 554 555 555	
27	109	2 555 556 555	
27	110	2 555 557 555	
27	111	3 555 558 555	
27	112	3 555 502 555	
27	113	3 555 503 555	
27	114	3 555 504 555	
27	115	2 555 500 582	
28	116	7 201 562 581	Trans. $j\omega_n^m$ to punch
28	117	7 201 564 583	Trans. e_n^{m-1} to punch
28	118	7 300 580 583	Punch σ_n, $j\omega_n^m$, K, e_n^{m-1}
29	119	-2 901 580 532	Trans. σ_n to σ_{n-1}
29	120	7 201 562 561	$j\omega_n^m$ to $j\omega_{n-1}^m$

Table 14. Continued

Program for Root-Locus Solution of Compensated System

30	121	8	100 040 002	Trans to P. P. 2 forty times
	122	0	000 000 000	Stop
	123	4	420 000 050	500: z_1
		0	000 000 000	501:p_1
		1	010 000 051	502:p_2
		1	290 000 051	503:p_3
		2	320 000 051	504:p_4
		2	000 000 050	505:2. 0
	124	5	000 000 049	506:0. 5
		2	000 000 049	507:0. 2
		3	141 592 750	508: π
		-2	000 000 050	509:-2. 0
		1	000 000 050	510:1. 0
		3	000 000 050	511:3. 0
	125	0	000 000 000	561:0. 0
	126	0	000 000 000	564;0. 0
	127	000 in columns 7, 8, 9 and 10		

Table 15.

Program for Gain Solution of Compensated System Along Real Axis

Box No.	Card No.	Order	Remarks
	01	0199 in columns 77, 78, 79 and 80	Problem No.
	02	9 800 001 000	Program Point 1
1	03	1 500 508 500	Compute K
2	04	-1 501 500 520	
2	05	-1 503 500 521	
2	06	-1 504 500 522	
2	07	-1 505 500 523	
2	08	2 500 521 521	
2	09	2 521 522 521	
2	10	2 521 523 521	
2	11	2 521 501 521	
2	12	3 521 520 521	
2	13	3 521 503 521	
2	14	3 521 504 521	
2	15	3 521 505 525	
3	16	7 201 500 524	Move σ_n to σ_{n-1}
3	17	7 300 524 525	Punch σ_{n-1}, K
4	18	8 100 021 001	Trans. to P.P. 1 21 times
	19	9 800 002 000	Program Point 2
5	20	1 506 508 534	Compute σ_n
6	21	-1 505 534 527	Compute K
6	22	-1 504 534 528	
6	23	-1 534 503 529	
6	24	-1 534 501 530	
6	25	2 534 527 527	
6	26	2 527 528 527	
6	27	2 527 529 527	

Table 15. Continued

Program for Gain Solution of Compensated System Along Real Axis

Box No.	Card No.	Order	Remarks
6	28	2 527 501 527	
6	29	3 527 530 527	
6	30	3 527 503 527	
6	31	3 527 504 527	
6	32	3 527 505 535	
7	33	7 201 534 506	Move σ_n to σ_{n-1}
7	34	7 300 534 535	Punch σ_n , K
8	35	8 200 014 002	Trans. to P. P. 2 14 times
	36	9 800 003 000	Program Point 3
9	37	1 507 509 544	Compute σ_n
10	38	-1 544 505 537	Compute K
10	39	-1 544 504 538	
10	40	-1 544 503 539	
10	41	-1 544 501 540	
10	42	2 544 537 545	
10	43	2 545 538 545	
10	44	2 545 539 545	
10	45	2 545 501 545	
10	46	3 545 540 545	
10	47	3 545 503 545	
10	48	3 545 504 545	
10	49	3 545 505 545	
11	50	7 300 544 545	Punch σ_n , K
11	51	7 201 544 507	Trans. σ_n to σ_{n-1}
12	52	8 300 020 003	Trans. to P. P. 3 20 times
	53	0 000 000 000	Stop
	54	0 000 000 000	500:0. 0
		4 420 000 050	501:z_1
		0 000 000 000	502:p_1
		1 010 000 051	503:p_2
		1 290 000 051	504:p_3
		2 320 000 051	505:p_4

Table 15. Continued

Program for Gain Solution of Compensated System Along Real Axis

Box No.	Card No.	Order	Remarks
	55	1 010 000 051	506:p_2
		2 320 000 051	507:p_4
		2 000 000 049	508:0.2
		1 000 000 050	509:1.0
	56	0000 in columns 7, 8, 9 and 10	

Table 16.

Program for Root-Locus Solution with Parallel T Compensating Network

Box No.	Card No.	Order	Remarks
	01	0199 in columns 77, 78, 79, and 80 Problem No.	
1	02	3 502 504 520	Compute α_1
2	03	-1 502 520 521	e_1
2	04	-1 500 520 522	
2	05	-1 503 520 523	
2	06	3 901 520 524	
2	07	3 901 521 525	
2	08	3 901 522 526	
2	09	3 901 523 527	
2	10	1 524 526 528	
2	11	-1 528 525 528	
2	12	-1 528 527 528	
3	13	1 520 505 530	Compute α_2
	14	9 800 001 000	Program Point 1
4	15	-1 502 530 521	Compute e_2
4	16	-1 500 530 522	
4	17	-1 503 530 523	
4	18	3 901 530 524	
4	19	3 901 521 525	
4	20	3 901 522 526	
4	21	3 901 523 527	
4	22	1 524 526 529	
4	23	-1 529 525 529	
4	24	-1 529 527 529	
5	25	-2 520 529 534	Compute α_{K+1}
5	26	4 528 530 534	
5	27	-1 528 529 535	
5	28	3 534 535 531	
6	29	7 300 528 531	Punch $e_{K-1}, e_K, \alpha_K, \alpha_{K+1}$
7	30	7 201 529 528	Transfer e_K to e_{K-1}

Table 16. Continued

Program for Root-Locus Solution with Parallel T Compensating Network

Box No.	Card No.	Order				Remarks
7	31	7	201	530	520	Transfer α_K to α_{K-1}
7	32	7	201	531	530	α_{K+1} to α_K
8	33	8	845	529	001	Trans. to P. P. 1 until $e_K < 10^{-5}$
	34	7	201	530	561	Trans. α_m to "561"
	35	9	800	002	000	Program Point 2
9	36	1	536	505	580	Compute $j\omega_n$
	37	9	800	003	000	Program Point 3
10	38	1	561	506	562	Compute σ_n^K
	39	9	800	004	000	Program Point 4
11	40	-1	503	562	537	Compute
11	41	-1	502	562	538	$\pi - \phi_1$
11	42	-1	500	562	539	ϕ_3
11	43	3	580	562	540	
11	44	3	580	537	541	ϕ_2
11	45	3	580	538	542	θ_1
11	46	3	580	539	543	
11	47	0	305	540	544	
11	48	0	305	541	545	
11	49	0	305	542	546	
11	50	0	305	543	547	
12	51	1	544	547	565	Compute e_n^K
12	52	-1	565	545	565	
12	53	-1	565	546	565	
13	54	7	201	561	560	Transfer σ_n^{K-1} to σ_n^{K-2}
13	55	7	201	562	561	σ_n^K to σ_n^{K-1}
13	56	7	201	564	563	e_n^{K-1} to e_n^{K-2}
13	57	7	201	565	564	e_n^K to e_n^{K-1}
14	58	7	300	560	564	Punch σ_n^{K-1}, σ_n^K, e_n^{K-1}, e_n^K
15	59	8	700	510	005	Test "510" to P. P. 5 if pos.
16	60	-1	564	563	509	Compute d_n^K

Table 16. Continued

Program for Root-Locus Solution with Parallel T Compensating Network

Box No.	Card No.	Order				Remarks
17	61	8	700	509	003	Test "509" to P. P. 3 if pos.
	62	9	800	005	000	Program Point 5
18	63	-1	508	901	508	Subtract one from "508"
18	64	1	510	901	510	Add one to "510"
20	65	8	700	508	003	Test "508" to P. P. 3 if pos.
21	66	-2	560	564	567	Compute σ_n^{K+1}
21	67	4	561	563	567	
21	68	-1	563	564	568	
21	69	3	567	568	562	
22	70	8	845	564	004	Trans. to P. P. 4 until $e_n^K < 10^{-5}$
23	71	-1	503	562	550	Compute K
23	72	-1	502	562	551	
23	73	-1	500	562	552	
23	74	2	562	562	549	
23	75	2	550	550	550	
23	76	2	551	551	551	
23	77	2	552	552	552	
23	78	2	580	580	553	
23	79	1	553	549	549	
23	80	1	553	550	550	
23	81	1	553	551	551	
23	82	1	553	552	552	
23	83	0	300	549	549	
23	84	0	300	550	550	
23	85	0	300	551	551	
23	86	0	300	552	552	
23	87	2	549	550	550	
23	88	2	550	551	550	
23	89	2	550	500	550	
23	90	3	550	550	550	

Table 16. Continued

Program for Root-Locus Solution with Parallel T Compensating Network

Box No.	Card No.	Order	Remarks
23	91	3 550 502 550	
23	92	3 550 503 581	
24	93	7 201 562 579	Move σ_n^m to Punch
24	94	7 201 564 582	Move e_n^m to Punch
24	95	7 300 579 582	Punch $\sigma_n^m, j\omega_n, K, e_n^m$
25	96	7 201 580 536	Trans. $j\omega_n$ to $j\omega_{n-1}$
25	97	7 201 562 561	σ_n^m to σ_{n-1}^m
26	98	8 100 060 002	Trans. to P. P. 2 60 times
	99	0 000 000 000	Stop
	100	1 800 000 051	500:z_1
		0 000 000 000	501:p_1
		1 260 000 051	502:p_2
		2 320 000 051	503:p_3
		2 000 000 050	504:2.0
		5 000 000 049	505:0.5
	101	2 000 000 049	506:0.2
		0 000 000 000	507:0.0
		3 000 000 050	508:3.0
		1 000 000 050	509:1.0
		-2 000 000 050	510:-2.0
	102	0 000 000 000	534:0.0
	103	0 000 000 000	564:0.0
	104	0000 in columsn 7, 8, 9 and 10	

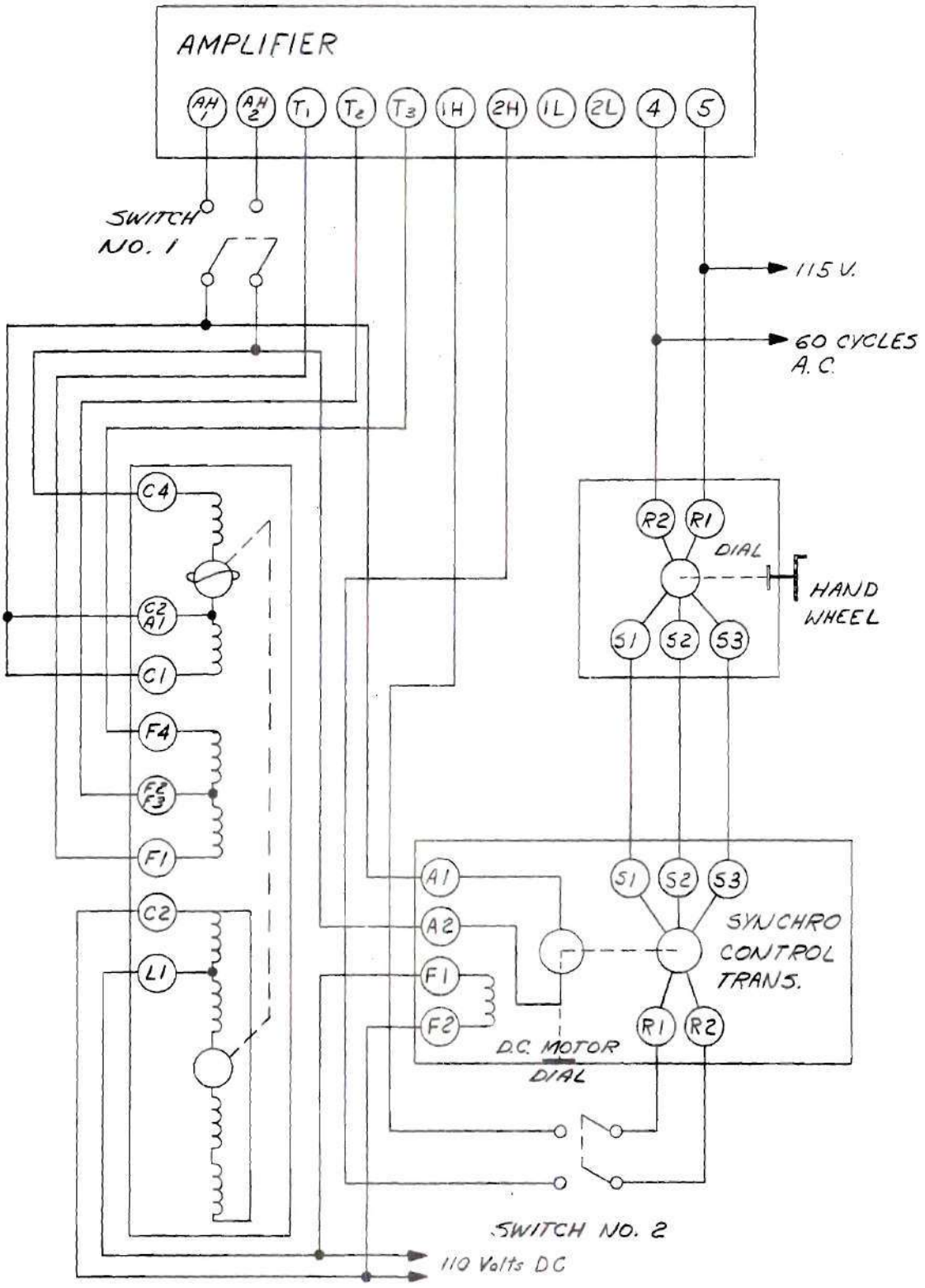

Figure 32. Schematic Diagram of Amplidyne Servo System.

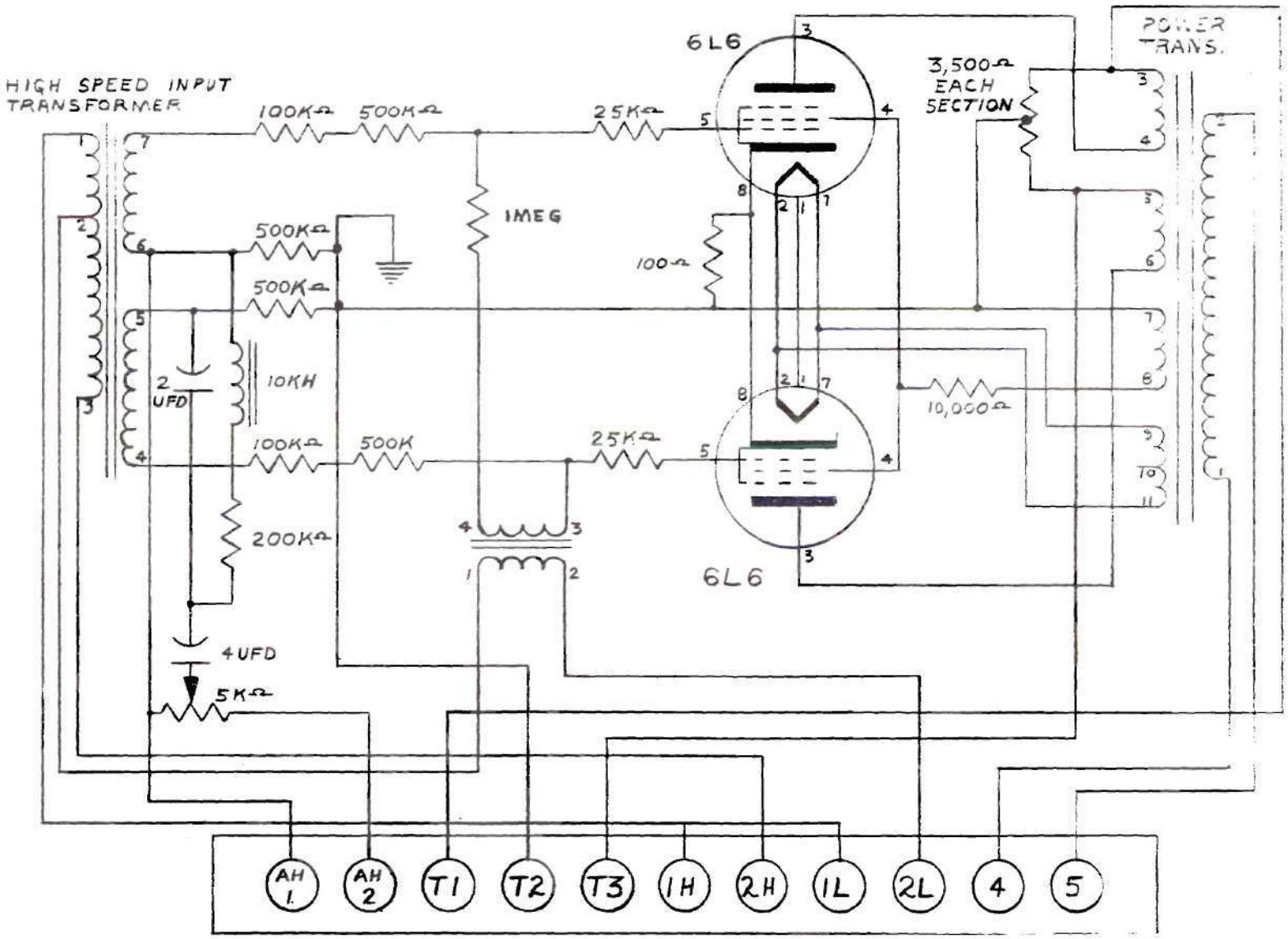

Figure 33. Amplifier Schematic Diagram.

BIBLIOGRAPHY

1. Evans, W.R., "Graphical Analysis of Control Systems", Transactions of the American Institute of Electrical Engineers, Vol. 67, 1948, pp. 547-551.

2. Evans W. R., Control System Dynamics, New York: McGraw-Hill Book Co., 1954, pp. 122-185.

3. Truxal, J.G., Control System Synthesis. New York: McGraw-Hill Book Co., Inc., 1955, pp. 227-231.

4. Evans, W.R., Control System Dynamics. New York: McGraw-Hill Book Co., Inc., 1954, pp. 237-241.

5. Brown, G.S. and D.P. Campbell, Principles of Servomechanisms. New York: John Wiley and Sons, Inc., 1948, pp. 236-251.

6. Truxal, op. cit., pp. 348-356.

7. Thaler, J.G. and R.G. Brown, Servomechanism Analysis. New York: McGraw-Hill Book Co., Inc., 1953, pp. 240-246.

8. Truxal, op. cit., pp. 345-348.

9. James, H.M., N.B. Nichols and R.S. Phillips, Theory of Servomechanisms. New York: McGraw-Hill Book Co., Inc., 1947, pp. 117-130.

10. Blanton, H.E., "Carrier Compensation for Servomechanisms", Journal of the Franklin Institute, Vol. 250, 1950, pp. 391-407.

11. Ibid., pp. 525-542.